LA-CC #	C#	Software Title	Software Description	Date
13-001	C13073	LANL Speech Processing Software	This software implements the MIMICRI algorithm for processing speech, as well as a variety of other standard speech processing algorithms. MIMICRI code is used to find transformations of signals so the signals will lie on bandlimited paths, to cross-validate the results of using the transformation, and to find the bandlimited paths once the transformations have been found. This is applicable to signal processing tasks such compression, speech recognition, speaker recognition, speech synthesis, etc.	1/15/2013
13-002	C13072	Randomized_Select_GPU 1.0_publicly available	This code is an implementation of a randomized selection algorithm, implemented on the GPU using CUDA. The description of the basic selection problem (but not this algorithm) is found in Art of Computer Programming, vol. 3: Sorting and Searching" by Donald Knuth.	1/15/2013
13-003	C13074	TWEPI (Twitter for Epidemic Analysis) 0.1	This software is a package for acquiring and analyzing data from the social internet. Currently, the software deals only with Twitter, but in the future we plan to extend this to other systems (e.g., Wikipedia, blogs, etc.). The software has or will have the following capabilities: 　Acquire and preprocess data (e.g., messages) from social internet systems. 　Extract geographic information from these data and infer missing geographic information. 　Estimate the demographics of social internet users. 　Estimate parameters useful in epidemic modeling (e.g., hand washing rate, school absenteeism, public transportation use). Future work may include extending the software to have additional related capabilities.	1/15/2013
13-004	C13075	High Magnetic Field Research Software Library, Version 1	High Magnetic Field Research Software Library is a collection of tools created to acquire and analyse experimental data at the Pulsed Filed Facility of the National High Magnetic Filed Laboratory.	1/17/2013
13-005	C13076	ionpython.py 2.X	This python code can analyze deep sequencing data from 454, Ion Torrent and Miseq sequencers. It finds all the HCDR3s present in the library and bins them by abundance.	1/17/2013
13-006	C13077	VPSC-ABAQUS UMAT 1.0	The VPSC-ABAQUS UMAT is an user-defined material subroutine for the commercial ABAQUS-Standard Finite Element package. The UMAT is based on the ViscoPlastic Self-Consistent (VPSC) formulation and the corresponding stand-alone VPSC code. The VPSC is a multipurpose polycrystal plasticity FORTRAN code, based on the knowledge of the mechanisms of slip and twinning that are active in single crystals of arbitrary symmetry and the texture of the aggregate. VPSC can be used to predict the effective stress-strain response, texture evolution, anisotropy, etc., and as a predictive tool for parameter identification, interpretation of experimental results and multiscale calculations, for academic and industrial applications to metallic, geologic and polymeric materials.	1/23/2013
13-007	C13078	pyHarmonySearch, version 1.0	I have implemented the harmony search algorithm (http://en.wikipedia.org/wiki/Harmony_search) in Python. The algorithm is a generic heuristic search algorithm, similar in nature to simulated annealing, tabu search, and evolutionary algorithms. The goal is to vary parameter input in some way so as to minimize or maximize the result.	1/29/2013

13-008	C13079	HMMDS3, version 1.0	HMMDS3 provides python3 code that implements the following algorithms for hidden Markov models:	1/29/2013
			Forward: Recursive estimation of state probabilities at time t, given observation likelihoods for times 1 to t	
			Backward: Combined with Forward, provides estimates of state probabilities at each time given all of the observation likelihoods.	
			Train: Implements Baum Welch algorithm which finds local maximum of likelihood of model parameters.	
			Decode: Implements Viterbi algorithm for finding most probable state sequence.	
			Implementations of the above algorithms are independent of the observation model. HMMDS3 enables users to implement any observation model by writing code for a class that provides methods for calculating the likelihood of an observation given a state and for reestimating model parameters given observations and state likelihoods.	
			HMMDS3 includes implementations of the following observation models:	
			Discrete: Integers in a finite range	
			Gauss: Floats with state dependent mean and variance	
			Class_Y: Observations that can include classification data	
13-009	C13080	Multi-Instrument Collect, 2.0.0.8	The Multi-Instrument Collect, MIC, program is designed for unattended collection and saving of data from multiple, distributed data acquisition legacy Safeguards instruments.	1/29/2013
13-010	C13081	Multiscale Plate Theory FEAP Finite Element	A finite element implementation of a variationally based, multiscale plate theory for FEAP.	2/25/2013
13-011	C13082	Maxentropic Transitions for Isentropic Expansion (METFIE), Version 1.x	Functions of the code METFIE version 1.0 (Maxentropic Transitions for Isentropic Expansion) 1. Calculates the maximum entropy transition matrix consistent with an adjacency matrix. 2. Calculates an adjacency matrix that characterizes the constraints described in the papers Propagating Uncertainty About Gas EOS to Performance Bounds for an Ideal Gun" LA-UR-12-22731 and "Constraining	3/5/2013
13-012	C13083	No Copy MOAB to VTK mini app, version 1.x	The "No Copy MOAB to VTK" mini app is software that compares visualization and analysis performance between a native MOAB mesh with run-time translation to a VTK mesh (shallow copy) versus a MOAB mesh copied to a VTK mesh (deep copy). MOAB is an open source mesh library found at http://trac.mcs.anl.gov/projects/ITAPS/wiki/MOAB VTK is an open source visualization and analysis library found at http://vtk.org/	3/6/2013
13-013	C13084	CONSIM v2.2	CONSIM is a framework for simulation and analysis in the United States NuDet Detection System (USNDS). It provides infrastructure for management of simulated constellations, sensors, NuDets, and backgrounds. It can function in either event-driven or continuous-time modes; different sensors and phenomenologies are amenable to different approaches. Physics propagation and source models are available for hard photon (x-ray, gamma-ray), neutron, optical, and EMP signals.	3/6/2013
13-014	C13085	TravelingSalesman GPU Version 1.0	The software is an implementation of the classic Traveling Salesman Problem in Computer Science. It is designed as example of Java and CUDA code interaction. It implements two algorithms - a simulated annealing algorithm and a genetic algorithm. Each is implemented twice - once as a CPU-only version and once as a CPU/GPU version.	3/6/2013

| 13-015 | C13086 | Scout Language Compiler 2.X | Scout is a domain-specific programming language and development toolchain that can support existing scientific applications on emerging architectures without having to significantly rewrite or refactor code.
\tIn situ Scout code can do computation, numerical or visual analysis on the data without storing it to file or post-processing.

Scout’s conservative extensions to C/C++ currently provide:

  A computational mesh abstraction that supports 1, 2 and 3D-mesh elements including mesh members (fields) for cells, vertices and points.
  Parallel forall over meshes or arrays for general processing.
  Parallel renderall over meshes for visualization of 2 or 3D datasets.
  Access to mesh element neighbors via cshift operation.
  Filtering ability for parallel constructs via where clause.
  Support for two-, three-, and four-component vector types.
  Stand-alone or in situ Scout programs.
  Parallel constructs running on single or multiple CPU cores or GPU. | 3/6/2013 |
| 13-017 | C13097 | Etch 4.0 | Etch is an interactive multi-contour tabular surface editor. Using Etch, you can load or create an entire project's worth of contours at the same time. You can create a complete model right in the drawing window, watch your changes take immediate effect, and see how those changes affect the rest of the model.

Etch contains tools for creating simple contours (points, lines, and arcs). It also contains an extensive toolset for editing these contours. Edit functions include offset, refine, redistributed, reflect, split, join, round, chamfer, scale, translate and rotate. The code incorporates unique LANL algorithms created for Wilson-Fowler refine and geometric offset. It also contains a diagnostics test set for validating the contours.

Etch also has two distinct interfaces: (1) a graphical user interface (GUI) and (2) a scripting interface. Etch’s scripting interface that may be run from Python. The GUI is a modern, intuitive interface. Inspired by many popular editing tools, Etch looks, feels, and acts like a drawing program. Buttons have descriptive mouse-over tags, objects are right-clickable, toolbars and windows can be dragged into new positions on the screen (or turned off completely), and keyboard shortcuts help you do routine tasks with a single button push. The GUI may be used to view contours created in a python script or it may be used to generate the contours directly.

Etch 4 differs from Etch 3 in that the GUI no longer generates and stores python scripts. Instead it modifies the contours directly and stores the modified contours. It does not save the original contour nor does it directly track contour changes. Instead it saves sessions so that a user may revert to an early session. | 3/14/2013 |

| 13-018 | C13096 | Osito 1.3 | Osito is a newer version of OSO, a constructive solid geometry (CSG) editor created by LANL. Osito combines the old OSO rendering engine with a Qt Graphical User Interface.

Osito differs from OsoLoco in two ways, (1) it contains a simple tabular surface editor called, Stable, and it’s 100% C++. Stable contains all the necessary functions needed to view, create, and edit a tabular surface; however, only a single contour may be edited at any one time. Because Osito is C++, it does not require python and has been built for Linux and MacOS.

Osito is a constructive solid geometry (CSG) editor. It contains several different primitive surfaces like spheres, cylinders, and boxes. The area inside or outside these surface is defined by using unary operators such as less than (LT) or greater (GT). Then these areas are combined into more complex solid geometry using intersection (AND) and union (OR) operators. Solid geometry is referred to as regions.

If you've used OSO in the past, we've tried to keep much of the look and organization of the original software; however; it now contains a modern, intuitive interface. Buttons have descriptive mouse-over tags, objects are right-clickable, toolbars and windows can be dragged into new positions on the screen (or turned off completely), and keyboard shortcuts help you do routine tasks with a single button push.

Osito contains an additional feature that allows the user to slice a 3D object along the z-axis and create a 2D contour. Using this function the user may convert a 3D model to 2D.

Osito may be used to create, edit, or view a MCNP model. Osito does support most MCNP geometries including a plane, sphere, circular cylinder, cone, ellipsoid, torus, box, parallelepiped, hexagonal prism, truncated cone, wedge, arbitrary polyhedron, and XYZP. It does not handle all types quadratics, non-circular cones or cylinders, or lattice. Osito has a special MCNP mode. This mode will allow you to view MCNP surfaces, cells, materials and TR cards. You may also use MCNP universes to fill geometry. Osito also has dialog box for viewing the message block, title card, and other data cards but it will not evaluate data cards for correctness. | 3/14/2013 |
| 13-027 | C13098 | tio, version 1 | The C++ code provides an example of how to use threading within an MPI application to help improve I/O performance. The code uses OpenMPI for MPI and MPI I/O. It is used to write a single large file, which is then read in to test non-threaded versus threaded I/O read and broadcast. It uses the latest C++11 threading facility. | 3/14/2013 |

| 13-028 | C13099 | Unified Host Collection Agent, version 1 | The Unified Host Collection Agent framework is a collection of software components that work in concert to collect and archive host-centric event information on hosts running Linux, Mac OS X and Windows operating systems. The system has two primary software components: the host agents and the event server. The host agents run locally and collection data relevent to network security operations. Data is encrypted by the agent and periodically transmitted to the event server. The agents are extensible in the data they collect and currently collect information related to general host status, running processes and network connection state and activity. Event data is encoded as JavaScript Object Notation (JSON) data to provide network secuirty research, development and operations professionals an easy to process data source for both streaming and post-processing based applications.

Host agents generate data either via polling or event-driven collection. Event data is encoded as JSON data and encrypted with a shared key. The data is placed into a UDP packet to be sent to the event server. Each agent is derived from a common python codebase which is extended for use on the three currently suppported operating system platforms. Agents are deployed as an executable to be run as a background process or service on the host operating system. These executables are self-updating. The agent, while
running, monitors for notifcation that it needs to upgrade using the DNS system to signal an update event. Upon receiving this signal the agent will download the new executable and then restart itself running the latest version of the agent.

The event server is a UDP server applpication written in python. It processes data sent from host agents and writes the encoded data to a streaming data feed and a data archive. Data is decrypyted upon receipt and then written immediatly to disk for long term archival and for use in post-processing operations (i.e. map-reduce). Optionally, the data is sent to a Redis server allowing for stream-based systems to subscribe to live feeds of host agent data.

The framework contains a simple web interface used in testing for monitoring agent status and event data. A connector to the Splunk log management platform is included to transform JSON event data into a format that is easily integrated into Splunk. | 3/14/2013 |
| 13-029 | C13100 | HIP - Hyperspectral Image Processor   Version 1.0 | HIP is an acronym for Hyperspectral Image Processor. The code is written in C, C++, and TCL/TK, and can be compiled on Windows, and Linux. The HIP code processes data from hyperspectral image sensors in the wavelength bands LWIR and MWIR. Recently it has been updated to process hypertemporal data. HIP provides a detailed analysis of the data from a variety of remote sensors that have been developed over the last 15 years. With its capability to model hypertemporal data, it processes data from a number of different cameras.
The analysis includes:
1) modelling of the atmosphere
2) selection of a variety of chemical materials and other targets of interest to look for
3) least squares analysis using a whitening transform
4) Bayesian Model Averaging analysis
5) an inerative GUI that permits detailed viewing of the data | 3/19/2013 |

13-030	C13101	LOBO - Long Wave Onboard Processor Version 1.4	LOBO is an acronym for Long Wave Onboard Processor. The code is written in C, C++, and TCL/TK, and is compiled on Windows. The LOBO code processes data on a platform (airplane) from hyperspectral image sensors in the LWIR wavelength band as the sensors process data. After a flight the LOBO analysis provides a cue report that guides further analysis on the ground. LOBO provides a detailed analysis of the data from a variety of remote sensors that have been developed over the last 15 years. The analysis includes:
			  1) For some sensors, calibration of raw data.
			  2) modelling of the atmosphere
			  3) selection of a variety of target chemicals and materials to look for,
			  4) least squares analysis using a whitening transform
			  5) Bayesian Model Averaging analysis
			  6) an iterative GUI that permits detailed viewing of the data,
			  7) automated processing of large data collections with no intervention by the operator.
				3/19/2013

13-031	C13102	VNIR-SWIR Exploitation Software Suite Version v18	The VNIR-SWIR exploitation software is a suite of tools that analyzes hyper-spectral radiance data cubes in the VNIR-SWIR wavelength range of the electromagnetic spectrum for specific reflectance signatures and assigns confidence levels to each detected region of interest.	3/19/2013
13-032	C13104	Random123 1.07	The copyrightable work is a set of patches to an open-source library of random number generators called Random123, which was developed by D. E. Shaw Research. The patches add support for two new compilers (Portland Group and IBM) and one new architecture (Blue Gene/Q).	3/19/2013
13-033	C13103	dns_parse_2.0	Dns_parse takes pcap of dns data, parses it and outputs it in a trivially re-parsable format. It handles dns sent over all major lower level protocols (Ethernet, IPv4, IPv6, UDP, TCP). Most major DNS subprotocols are handled, and for those that aren't the raw data is printed in an easily reconstructed format.	3/19/2013
13-034	C13105	LanlApp	The software is an IOS/Android Native application that will be used to highlight the Lab's 70th.	3/21/2013
13-035	C13106	pysysdevel, 0.5.2	pysysdevel is a Python package that assists software development in three related areas: model development, data visualization, and collaboration & distribution. This package facilitates multi-model simulation development by implementing a generic Model-View-Controller interface. The View portion of this interface handles data visualization by unifying both web-based and desktop UI tools. For package distribution, pysysdevel extends the distutils package to handle external libraries and application freezing.	4/4/2013
13-036	C13107	Message Disembogulator Suite (MDS) 1.X	The Message Disembogulator Suite (MDS) is a set of tools and simulators to estimate the performance of supercomputer/cluster interconnects. These tools are used to compare different topologies of various scales, such as fattree, dragonfly, mesh, tori, and hypercubes. The simulators are of various abstractions, such as flow based simulation, which looks a infinite length messages and calculates the utilization of the links in the interconnect, and high level analytical performance calculators which look at average link utilization based on communication patterns. The communication patterns fed into the simulators can either be generated patterns such as random or nearest-neighbor or they can come from application traces. The tool set also include tools to extract traces from MPI applications, generate patterns, and tools that deal with the process to node mapping .	4/23/2013
13-037	C13108	FLAG, Version 3.3	FLAG is an ALE radiation hydrodynamics code. Code is constructed on a 1D/2D/3D aritrary polyhedral mesh based on domain decomposed MPI. FLAG can either import mesh based on X3D format or self generate a mesh. The hydrodynamics is staggered grid or cell centered with an operator split flux-based remap. The hydro includes the ability to model contact surfaces. The code supports a programmed mesh refinement capability and free lagrange formulations. The code handles multiple equations of state that include analytic, sesame, leos, and high explosives. Material models also include strength (PTW, KOSPALL,MTS), failure and damage (JCD,pmin,tensile). High explosive lighting models include direct, lund, DSD, and reactive flow. The radiation is grey diffusion with a 3 temperature (electron, ion, and radiation) treatment.	4/23/2013
13-038	C13109	Ingen version 2.4	Ingen is a general-purpose geometry modeling, mesh generation, and input specification software system. It is primarily targeted at geometry and meshing for 2-D cylindrically-symmetric hydrodynamic simulations using the ASC integrated physics codes. The code also supports creating geometry for 1-D, 2-D Cartesian, and 3-D simulations. Ingen does not perform the actual physics calculations. The system also has some capability for managing source data (e.g., contour data) and the execution of the target simulation codes. The system is a library with an interface designed for use in both client frameworks and simple single-problem scripts. The system includes a read-only graphical geometry and mesh viewer. For a complete list of functions see the code manual included herewith.	4/23/2013

13-039	C13110	RELIC (Rare Earth and Level Intensity Calculations), Version 1.x	RELIC stands for “Rare Earth Level and Intensity Calculation” and is a software developed by Markus Hehlen (MST-7, LANL). The software allows for the calculation of the energy levels of tripositive rare earth ions as well as oscillator strengths of respective 4f-4f transitions. It includes lower level functions for the calculations of basic quantum mechanical quantities (such as Wigner coefficients and reduced matrix elements). The higher level functions allow the user to fit the electrostatic and spin-orbit coupling parameters to a set of experimental 4f energy levels as well as to fit the Judd-Ofelt parameters to a set of experimental 4f-4f oscillator strengths. The software also calculates derived quantities such as excited state lifetimes and branching ratios. RELIC builds on quantum-mechanical formalisms that have been published in the literature and have recently been reviewed by M.P. Hehlen et al [1]. RELIC’s graphical user interface is a new development by the author. [1] M.P. Hehlen, M.G. Brik, K.W. Krämer, J. Lumin. 136 (2013) 221-239	4/23/2013
13-040	C13111	PeakEasy 4.5	PeakEasy is a gamma-ray spectroscopy tool intended for quick conformation and isotopic source reference. PeakEasy is designed around three isotope libraries, bringing together several interactive spectroscopy tools into one package. Along with a comprehensive isotope library, searchable database, and comparison spectra library, PeakEasy is able to load and display spectra data files from over 120 different file formats.	5/6/2013
13-041	C13123	Graphical Input Aggregate Control (GIAC), Version 23.x	Supporting Military Mission Rehearsals and Large Scale Simulation Training for over 20 years.\n \nGIAC is a perfect example of computer graphics, distributed computing, system configurations and software engineering being used in one complete project. GIAC is a software System comprised of 3 components: A Graphical User Interface (GUI) and GDS, a distributed data system, and a HLA interface(s). GIAC is over 700,000+ lines of custom C and X-Motif code. GIAC's main function is to display the electronic battle field, provide Situational Awareness (SA), using computer graphics, by the training simulations and war games as well as provide input (Command and Control - C2) for simulation objects in a distributed environment. GIAC also provides terrain analysis features such as Line of Sight (LOS) calculations, shaded relief mapping, radar echo analysis, terrain profiling and NGA image map displays. LOS calculations are done in a distributed manner. GDS allows exercises to be distributed. Distribution can be configured over a LAN, WAN, locally or worldwide. Distribution is accomplished via network programming using a socket api written by GIAC team members in a point to point scheme (TCP/IP). Both software components are a complex code using well known and customs written/modified algorithms/data structures to obtain the maximum performance and scalability/flexibility possible. GIAC is able to handle the distribution and interactions of over 900,000 simulation objects. The software can function on Multiple Hardware platforms (HP, SPARC, and Intel) under those systems native OS as well as Linux. GIAC operates in an ALSP, HLA and DIS environment (separately or simultaneously or any combination thereof). GIAC can display and interact with LIVE, VIRTURAL and CONSTRUCTIVE entities.\n\nGIAC also can display information gotten from Real World Systems. Right now, GIAC will display TADIL-J tracks and OTH-Gold tracks. GIAC can also dead-reckon objects (provide SA data of long periods of time (30 minutes) until an actual real world update is provide). GIAC also works over crypto gear. GIAC has an extensive filtering capability as well (able to filter objects and track by attribute and source - customize the SA if you will).\n\nGIAC is compatible with Google Earth, saving the military training community millions of dollars.\n\nMajor uses over the last 2-3 years have been for Home Land Defense, the Millennium Challenge 02 Experiment, UFL/UFG, all JNTC events, the Air Force's Distributed Mission Training and many Joint Task Force Level training exercises. Every Military Simulation Center the US has uses GIAC as well as NATO. The GIAC software is use 24/7/365. GIAC software itself is completely unclassified. \n \nSpecifically for Millennium Challenge 02 (MC02) and JNTC, GIAC was modified to show all the model entities in the MC02 and JNTC Federations minus single infantry men in a WHITE CELL (ground truth) mode. GIAC was also used as a debugging tool for the federation. Also, GIAC is the main C2 control for AWSIM object in both perceived and ground truth modes. GIAC allowed AWSIM operators to target federation objects and provided a mission history for MISREPS and BDA reports.	5/23/2013

| 13-042 | C13118 | Magpie-ec-libs, Version 0.1 | The Magpie software is a framework for the calculation of thermodynamic state and properties for materials (i.e. components) and mixtures of materials, using complete thermodynamic equations of state. Certain component models depend on intermolecular potentials, and/or on forms of Gruneisen gamma. Magpie thus contains a library of component models, mixture models, intermolecular potentials, and Gruneisen gamma.

The magpie-ec-libs package contains only the parts of the equation of state library and the records in the material library that are restricted to export. (Most are covered under ITAR category XVI, though export restrictions could also be triggered for other reasons.)

Component equation of state (EOS) models available in magpie-ec-libs version 0.1:
--Ross perturbation theory with arbitrary, spherical, intermolecular potential form
--P-alpha compaction model with arbitrary solid EOS
--Hayes
--Sesame decomposition
--Debye Ionic
--Einstein Ionic
--Thomas-Fermi Electronic
--Thomas-Fermi-Dirac Electronic
--Vinet Cold Curve
--Murnaghan Cold Curve

Intermolecular potential forms available in magpie-ec-libs version 0.1:
NONE

Mixture models available in magpie-ec-libs version 0.1:
NONE

Gruneisen gamma forms available in magpie-ec-libs version 0.1:
NONE

Specific materials included in the magpie-ec-libs material library version 0.1:
--Ar | 5/21/2013 |
| 13-043 | C13112 | pubmed 1.0 | R code to download XML documents from the Open Access subset of NCBI's Pubmed Central and search for microbial gene locus tags within full text or in structured tables in text and supplements. | 5/8/2013 |

13-044	C13113	Magpie-base, Version 0.1	The Magpie software is a framework for the calculation of thermodynamic state and properties for materials (i.e. components) and mixtures of materials, using complete thermodynamic equations of state. Certain component models depend on intermolecular potentials, and/or on forms of Gruneisen gamma. Magpie thus contains a library of component models, mixture models, intermolecular potentials, and Gruneisen gamma. The Magpie-base package contains only those equation of state models, and material library entries that are unrestricted to export. Component equation of state (EOS) models available in magpie-base version 0.1: --ideal gas --stiffened-gas --hard-sphere (Carnahan-Starling) --Polytropic gas --Birch-Murnahan cold curve Intermolecular potential forms available in magpie-base version 0.1: --exponential-6 --Lennard-Jones Mixture models available in magpie-base version 0.1: --ideal mixing Gruneisen gamma forms available in magpie-base version 0.1: --constant gruneisen gamma --constant value of gruneisen gamma multiplied by density In addition to specific physics models, the magpie-base package contains the required infrastructure to calculate thermodynamic curves for a general model, namely: --isobar --isochore --isotherm --isentrope	5/8/2013
13-045	C13114	STOW (Security Tool on Windows) Version 2.1	Security Tool in the form of a Group Policy Object or GPO attached to Active Directory.	5/8/2013
13-046	C13117	RealBogus, Version 3.x	RealBogus 3.x is a Python package for machine learning classification of candidate variable objects detected by modern imaging sky monitoring surveys. It was developed and tuned using data from Palomar Transient Factory (PTF) survey. Highly accurate classification is obtained using the random forest algorithm. The package includes functionality to read training data, modify configuration parameters, train the classifier, store a trained classifier to a file and load it back, apply a trained classifier to new data. An important feature of this code is the ability to perform periodic updates to continuously growing database of variability candidates. Results are reported as RB3 scores (probabilities) and written back to the database as the new data arrives.	5/21/2013
13-047	C13115	Model for Prediction Across Scales - Ocean, Version 0.0.0	Software produces simulations of the ocean system. Capabilities include simulation of idealized test cases and real-world configurations. Intended use is for the scientific study of the ocean system and global climate change. The numerical method is finite-volume based on knowledge available in the peer-reviewed literature.	5/8/2013

13-048	C13116	LANL contributions to astrophysics simulation codes such as GR1D, Zelmani, and SpEC.	This project involves the addition of improved numerical methods to existing codes, such as Caltech's open-source GR1D code (see stellarcollapse.org: GR1D - A New Open-Source Spherically-Symmetric Code for Stellar Collapse to Neutron Stars and Black Holes); SpEC, a 3D general relativistic spectral code maintained by the Simulating eXtreme Spacetimes (SXS) collaboration between Caltech and Cornell (not open-source, see http://black-holes.org/SpEC.html) used for simulating compact object mergers; and Zelmani (see www.stellarcollapse.org/ottetal2012b) for 3D simulations of core collapse. In these codes, a two-moment aproach to simulate the neutrino transport can be used, where the 0th and 1st moment equations are solved and then they need to be closed with an analytic approximation of the Eddington tensor from the 2nd moment. Various approximations will be investigated and compared to Monte Carlo calculations as a benchmark for calibration. Based on the experiences gained in 1D with GR1D, the models will be extended to 3D codes such as SpEC and Zelmani.	5/8/2013
13-049	C13119	Gazebo, 1.X	Software framework for submitting and analyzing system tests.	5/21/2013
13-050	C13120	AEM Realtime Monitor, Version 1 (Publicly available)	AEM Realtime Monitor abstract The AEM Realtime Monitor software reads files containing spectral data produced by the EM_DAC enrichment monitoring service, as well as files containing pressure data produced by the Serial Datalogger software, and processes the data contained in them to produce results that reflect the enrichment of the UF_6 gas in the header pipe to which the detectors are attached. The software displays the data in near-real time, and as averages across several time ranges. It provides the ability to export raw data and enrichment results to Excel-compatible files and to find new values for the constants used to correct for wall deposits in the header pipe.	5/21/2013
13-051	C13122	CBTF contrib LANL Tools, Version 1.0	The CBTF tools created at LANL are added to the contrib section of the CBTF release. The tools are both system administration and user tools. The tools make use of the CBTF framework and the underlying MRNet network in order to be scalable and automatically adjust to new topologies. Currently there are five tools, two admin tools called psTool and tbonFS, and 3 user tools called memTool repeatingMemTool and stack. The psTool will run ps on all backend nodes then use filters to identify the processes running on all nodes that are the same and what processes are unique to each node. The tbonFS tool is an implementation of TBON-FS for MRNet, it performs group file operations on all the backend node local file systems. It can run grep, tail, read or write files on the backend. MemTool is a user tool which can take a snapshot of the memory usage for a parallel application. The repeatingMemTool will take continuous snapshots of the memory usage for a parallel application and return a summary of memory usage. The stack tool will run gstack on each pid of a parallel application then perform cluster analysis on the stack traces in order to group similar stack traces together.	5/23/2013
13-052	C13121	Idealizer (C, Fortran, R, Java), Version 0.1	Data recordings often include high-frequency noise and baseline fluctuations that are not generated by the system under investigation, and which need to be removed before analyzing the signal for the system's behavior. In the absence of an automated method, experimentalists fall back on manual procedures for removing these fluctuations, which can be laborious and prone to subjective bias. We introduce a maximum likelihood formalism for separating signal from a drifting baseline plus noise, when the signal takes on integer multiples of some value, as in ion channel patch-clamp current traces. Parameters such as the quantal step size (e.g., current passing through a single channel), noise amplitude, and baseline drift rate can all be optimized automatically using the EM (expectation-maximization) algorithm taking the number of open channels (or molecules in the on" state) at each time point as a hidden variable. Our goal here is to reconstruct the signal, not model the (possibly highly complex) underlying system dynamics. Thus our likelihood function is independent of those dynamics. This may be thought of as restricting to the simplest possible hidden Markov model for the underlying channel current in which successive measurements of the state of the channel(s) are independent. The resulting method is comparable to an experienced human in terms of results, but much faster. Fortran 90, C, Java, and R codes that implement the algorithm are available for download from our website"	5/21/2013
13-053	C13126	pyxDamerauLevenshtein, Version 1.0	I implemented the Damerau-Levenshtein edit distance algorithm in Cython, a library which allows translation of Python code to C for high performance. This implementation runs 2 orders of magnitude faster than the fastest native Python implementation.	5/24/2013

13-055	C13127	Command, Control, & Telemetry Interface Software, Version 2	The Command, Control, & Telemetry Interface (CCTI) software is a suite of programs that is used during the development, integration, testing, and operation of the SABRS satellite payload. The CCTI software provides functionality to 1) control operation of various spacecraft interfaces and emulators, 2) command operation of various payload subsystems both interactively or through the generation of 'command plans' for future execution, and 3) collect, display, and archive telemetry data generated by the payload. The CCTI software communicate with Ground Support Equipment (GSE) to operate a satellite payload directly (acting as a surrogate for the spacecraft) or with a Ground Segment to operate a satellite payload hosted on a spacecraft.	5/24/2013
13-056	C13125	OsoLoco, Version 1.3	OsoLoco is a newer version of OSO, a constructive solid geometry (CSG) editor created by LANL. OsoLoco combines the old OSO rendering engine with a Qt Graphical User Interface. We've also integrated another setup tool, Etch. Etch looks, feels, and acts more like a drawing program than the tabular surface editor. It allows you to manipulate multiple contours within one workspace and then import in the resulting tabular into OSO. OsoLoco is a constructive solid geometry (CSG) editor. It contains several different primitive surfaces like spheres, cylinders, and boxes. The area inside or outside these surface is defined by using unary operators such as less than (LT) or greater (GT). Then these areas are combined into more complex solid geometry using intersection (AND) and union (OR) operators. Solid geometry is referred to as regions. If you've used OSO in the past, we've tried to keep much of the look and organization of the original software; however; it now contains a modern, intuitive interface. Buttons have descriptive mouse-over tags, objects are right-clickable, toolbars and windows can be dragged into new positions on the screen (or turned off completely), and keyboard shortcuts help you do routine tasks with a single button push. OsoLoco contains an additional function which allows the user to slice a 3D object along the z-axis and create a 2D contour. Using this feature the user may convert a 3D model to 2D. OsoLoco may be used to create, edit, or view a MCNP model. OsoLoco does support most MCNP geometries including a plane, sphere, circular cylinder, cone, ellipsoid, torus, box, parallelepiped, hexagonal prism, truncated cone, wedge, arbitrary polyhedron, and XYZP. It does not handle all types quadratics, non-circular cones or cylinders, or lattice. OsoLoco has a special MCNP mode. This mode will allow you to view MCNP surfaces, cells, materials and TR cards. You may also use MCNP universes to fill geometry. OsoLoco also has dialog box for viewing the message block, title card, and other data cards but it will not evaluate data cards for correctness.	5/24/2013
13-057	C13128	AEM Realtime Monitor, Version 1.x	The AEM Realtime Monitor software reads files containing spectral data produced by the EM_DAC enrichment monitoring service, as well as files containing pressure data produced by the Serial Datalogger software, and processes the data contained in them to produce results that reflect the enrichment of the UF6 gas in the header pipe to which the detectors are attached. The software displays the data in near-real time, and as averages across several time ranges. It provides the ability to export raw data and enrichment results to Excel-compatible files and to find new values for the constants used to correct for wall deposits in the header pipe.	5/29/2013
13-058	C13129	MCq2HFB (Multi-Component Quasi 2D Hartree Fock Bogoliubov)	MCq2HFB is a research code for quantum systems. main code: bdg.f90 and hfb.f90 subroutines: buildCX.f90 ffte.f90 grid.f90 semiclassical.f90 interaction.f90 make_interaction.f90 nl2.f90 spatialgrid3.f90	6/21/2013

13-059	C13130	Model for Prediction Across Scales - Ocean, Version 1.0	Software produces simulations of the ocean system. Capabilities include simulation of idealized test cases and real-world configurations. Intended use is for the scientific study of the ocean system and global climate change. The numerical method is finite-volume based on knowledge available in the peer-reviewed literature.	6/21/2013
13-060	C13133	MdataGen.py	MdataGen.py reads raw data files generated by seismic, infrasound, overpressure and weather instruments and converts them into a Mdata file format. Raw seismic data is presented in SAC file format and seismic data is written to a Mdata File in miniSEED file format so the software includes a SAC to miniSEED converter. The other raw data files are read directly by the software.	7/23/2013
13-062	C13132	SafetyCinemaApp, Version1.0	A Mobile app for both IOS and Android devices that will represent a mobile access of the LANL SafetyCinema content.	7/18/2013
13-063	C13137	CGMF, Version 1.0	The CGMF code is based on two previously codes developed at LANL: CGM (LA-CC-11-018) and FFD (LA-CC-10-003). It performs Monte Carlo simulations of the decay of excited fission fragments by emission of prompt neutrons and gamma rays. The Hauser-Feshbach statistical theory of compound nuclear reactions is used to compute the emission probabilities at each step of the cascade. Monte Carlo histories are recorded and analyzed. Average prompt fission neutron multiplicity (PFNM), PFNM distribution P(nu), average PFNM as a function of fragment characteristics (A,Z) can all be extracted from CGMF calculations. Similar quantities can also be obtained for prompt gamma rays. In addition, n-n, n-g, and g-g correlations can be studied both in energy and angle.	7/29/2013

13-064	C13138	BISONVPSC.C, Version 1 (superseded by CODE-2014-4/C14006)	BISONVPSC.C is a user interface computer code that was written for the purpose of integrating Viscoplastic Self Consistent (VPSC) computer program of Los Alamos National Laboratory (LANL), USA with the finite element package of Idaho National Laboratory (INL), USA called MOOSE - Multiphysics Object Oriented Simulation Environment.
			
			This program brings all the capabilities of VPSC program to the finite element package MOOSE of INL which can be used to perform large scale engineering simulations relevant to material deformation. By using BISONVPSC.C, VPSC program can be integrated at each integration point of a finite element mesh used in MOOSE. By doing such arrangement, realistic simulations of plastic deformation and creep of materials can be performed.
			For example, the following can be achieved by using the above interface:
			
			  Metal forming simulations
			  Prediction of material properties
			  Simulation of creep under irradiation and non-irradiation conditions.
			
			Since this computer code is used inside MOOSE finite element package, the finite element mesh file, initial boundary conditions and time stepping options are provided by the MOOSE input file. The material properties, crystallographic texture and microstructure details of the material are read from VPSC input files. BISONVPSC.C program then integrates MOOSE and VPSC and communicates the values of variables between the two using state variables.
				7/29/2013

13-066	C13147	spammpack, Version 2013-06-18	This library is an implementation of the Sparse Approximate Matrix Multiplication (SpAMM) algorithm. It provides a matrix data type, and an approximate matrix product, which exhibits linear scaling computational complexity for matrices with decay. The product error and the performance of the multiply can be tuned by choosing an appropriate tolerance. The library can be compiled for serial execution or parallel execution on shared memory systems with an OpenMP capable compiler.	8/21/2013
13-067	C13146	iNSECT, Version 2013-06-03	This program suite implements a machine learning approach to classifying genomic sequences. By labeling a set of known nucleotide sequences, the program is trained and produces a data file which is used during the classification step to classify unknown sequences into the known classes of the training set.	8/16/2013

| 13-068 | C13139 | This procedure describes the recovery and assay of Ge-68 from Ga targets encapsulated in niobium. These targets may be irradiated at a variety of facilities around the world including cyclotrons and linacs operated at energies between 20 and 800 MeV. Multiple targets may be processed into a single batch to assure that the Ge-68 concentration of the final product meets the requirements of the end-user.

Ge-68 produced using this procedure is not intended for use as an Active Pharmaceutical Ingredient (API). | Recovery of Ge-68 from Irradiated Ga Metal Targets\n\nThe Isotope Production and Applications Program (IPAP) produces radioactive materials by chemically processing irradiated targets at the TA-48 Hot Cell Facility. The Isotope Program procedures describe how the program implements quality assurance standards and good manufacturing practices for the production of isotope products. This procedure is one of a group of procedures intended to implement the mandated requirements. | 8/5/2013 |
|---|---|---|---|---|
| 13-069 | C13140 | Isolation | Isolation and Purification of Sr-82 From Irradiated RbCl Targets\n\nThe Isotope Production and Applications Program (IPAP) produces radioactive materials by chemically processing irradiated targets at the TA-48 Hot Cell Facility. Some of these materials are distributed to pharmaceutical companies and are ultimately used in medical procedures. As such, they fall under the purview of FDA and international guidelines for good manufacturing practices as applied to active pharmaceutical ingredients (API). These mandate a set of quality assurance standards that applied when processing or handling such materials. This procedure is one of a group of procedures intended to implement the mandated requirements. | 8/5/2013 |
| 13-070 | C13141 | Strontium-82 produced by the Isotope Production and Applications Program (IPAP) is ultimately used in medical diagnostics and is considered an active pharmaceutical ingredient (API). The U.S. FDA (Food and Drug Association) regulates the quality of all APIs. Therefore this procedure follows cGMP (current Good Manufacturing Practice) as outlined by the FDA. | Isolation and Purification of Sr-82 From Irradiated Rb Metal Targets\n\nThe Isotope Production and Applications Program (IPAP) produces radioactive materials by chemically processing irradiated targets at the TA-48 Hot Cell Facility. Some of these materials are distributed to pharmaceutical companies and are ultimately used in medical procedures. As such, they fall under the purview of FDA and international guidelines for good manufacturing practices as applied to active pharmaceutical ingredients (API). These mandate a set of quality assurance standards that applied when processing or handling such materials. This procedure is one of a group of procedures intended to implement the mandated requirements. | 8/5/2013 |
| 13-072 | C13135 | EPRO_CRACKER | This software functions to extract Los Alamos National Laboratory (LANL) United States Nuclear Detection System (USNDS) instrument data embedded in the USNDS Integrated Correlation and Display System (ICADS) End Products (EPROS) and output the LANL instrument data into *.cdf formatted files. | 7/25/2013 |

13-073	C13148	isClassify, a matlab script for Importance Sampling associated with LA-UR-12-21904 on Network Uncertainty Quantification	Matlab script that implements importance sampling for estimation of network topology in support of the work described in the following abstract. Abstract Power grid operators decide where and how much power to generate based on the current topology and demands of the network. The topology can change as safety devices trigger (connecting or disconnecting parts of the network) or as lines go down. Often, the operator cannot observe these events directly, but instead has contemporary measurements and historical information about a subset of the line flows and bus (node) properties. This information can be used in conjunction with a computational model to infer the topology of the network. We present a Bayesian approach to topological inference that considers a bank of possible topologies. The solution provides a probability for each member in the model bank. The approach has two important features. First, we build a statistical approximation, or emulator, to the computational model, which is too computationally expensive to run a large number of times. Second, we use the emulator in an importance sampling scheme to estimate the probabilities. The resulting algorithm is fast enough to use in real time and very accurate. This paper has supplementary online material.	8/21/2013
13-074	C13149	PyDAC Extensions	This project details extensions to the PyDAC framework. PyDAC is a extremely-specialized programming framework for a specific implementation of FPGAs (field programmable gate arrays) developed by Dr. Ron Sass and his students at the University of North Carolina at Charlotte. Specifically, their approach creates two types of processing units - "green" (energy efficient, small, cheap, plentiful, special purpose) and "white" (power hungry, large, conventional) cores. These cores are implemented (currently) using open standards. PyDAC and the FPGA "code" exist outside of LANL and have existed for the past few years. This project extends the above work by adding reliability and fault-tolerance techniques to the green/white approach. Specifically, we are working to add redundant multi-threading and triple modular (voting) redundancy. We are also extending the scratchpad (local memory) of each green core so that work can be staged in a pipeline fashion. To test the reliability extensions, we are adding a trivially simple fault injector that modifies input data. This feature allows us to demonstrate the RMT and TMR techniques and then measure the impact.	8/21/2013
13-075	C13150	Legion - LANL Version 0.1	Modern parallel architectures have both heterogeneous processors and deep, complex memory hierarchies. We present Legion, a programming model and runtime system for achieving high performance on these machines. Legion is organized around logical regions, which express both locality and independence of program data, and tasks, functions that perform computations on regions. We describe a runtime system that dynamically extracts parallelism from Legion programs, using a distributed, parallel scheduling algorithm that identifies both independent tasks and nested parallelism. Legion also enables explicit, programmer controlled movement of data through the memory hierarchy and placement of tasks based on locality information via a novel mapping interface. Under this release LANL will be contributing to both the runtime and language level support.	8/21/2013
13-076	C13151	QUO, Version 0.1-rc2	High-performance computing (HPC) system architectures are evolving, and as a consequence so are the programming paradigms employed within parallel and distributed scientific applications. Until recently, most large coupled multi-physics applications at Los Alamos National Laboratory (LANL) relied almost exclusively on distributed memory parallelism facilitated by an implementation of the message passing interface (MPI) to run massively parallel simulations on networked clusters of multi-core shared-memory nodes. Now, various physics packages at LANL have moved or are considering moving towards hybrid programming models (e.g. MPI + OpenMP, MPI + pragma-based offload) to better utilize the computational resources on current and emerging HPC architectures. The rate at which the respective packages are moving towards hybrid programming models, however, varies at a per-project level and is creating an ecosystem of packages with differing process binding policy needs. Consequently, the need for run-time tuning of process binding policies throughout the execution of coupled multi-physics codes has become apparent. In this work, we present a software solution to support evolving process binding policies within coupled multi-physics and multi-phase codes comprising physics packages with varying binding policy needs or containing evolving computational phases that may benefit from run-time process binding changes, respectively.	8/21/2013

13-077	C13145	Truchas 2.8	Truchas is software for simulating metal casting processes currently in use in LANL and industrial foundries. The software targets simulating the entire casting process, beginning with the preheat of mold materials by electromagnetic induction, free surface flow of a molten alloy being poured into a mold, followed by the rapid initial cooling and solidification of that alloy, ending with the slower cooling of the solidified part down to room temperature. Truchas has realistic physical models for electromagnetic radiation, alloy solidification and resulting micro/macro segregation phenomena, free surface incompressible flow, heat transfer, complex three-dimensional geometric effects, microstructural nucleation and evolution, surface tension, solid/solid phase transition effects, and residual stress buildup and response. Truchas can be used by foundry people in understanding their casting processes, as well as aiding in the design of improved casting processes which lead to cast metal alloy parts which meet microstructural, geometrical, and property specifications. Because of the breadth and generality of the physical models Truchas embodies, it is also a very general engineering analysis tool for a wide variety of industrial and natural phenomena. Truchas is written in modern Fortran, primarily for high-performance parallel computing platforms, but the software is portable to any platform possessing a modern Fortran compiler (including PCs).	8/8/2013
13-078	C13143	SUBSCALE, Version 0.1	This is small 1D research code for multi-material flows which allows to deal with voids in these cells. This code is created to test some closure models.	8/5/2013
13-079	C13134	Biowatch Sampling Management System, version 2.0.9	Biowatch Sampling Management System is a software package that tracks chain of custody of physical elements/equipment of the Biowatch sampling program. It also has accumulation and handling data capabilities.	7/25/2013
13-080	C13144	Truchas 2.8 - GE	Truchas is software for simulating metal casting processes currently in use in LANL and industrial foundries. The software targets simulating the entire casting process, beginning with the preheat of mold materials by electromagnetic induction, free surface flow of a molten alloy being poured into a mold, followed by the rapid initial cooling and solidification of that alloy, ending with the slower cooling of the solidified part down to room temperature. Truchas has realistic physical models for electromagnetic radiation, alloy solidification and resulting micro/macro segregation phenomena, free surface incompressible flow, heat transfer, complex three-dimensional geometric effects, microstructural nucleation and evolution, surface tension, solid/solid phase transition effects, and residual stress buildup and response. Truchas can be used by foundry people in understanding their casting processes, as well as aiding in the design of improved casting processes which lead to cast metal alloy parts which meet microstructural, geometrical, and property specifications. Because of the breadth and generality of the physical models Truchas embodies, it is also a very general engineering analysis tool for a wide variety of industrial and natural phenomena. Truchas is written in modern Fortran, primarily for high-performance parallel computing platforms, but the software is portable to any platform possessing a modern Fortran compiler (including PCs). This version of Truchas entitled Truchas 2.8 - GE is specifically for license to GE under a proprietary/controlled license and will not be released under any type of open source license.	8/5/2013
13-081	C13153	PyFEHM, Version 1.0 - 1.X	PyFEHM is a set of Python libraries designed to be used with the LANL simulation code FEHM. PyFEHM is primarily designed for the construction of FEHM input and grid files. Python capabilities include 1. Construction of orthogonal grids or arbitrary complexity. Grid visualisation. 2. Construction of FEHM input files supporting a range of macros including perm, rock, cond, zone, pres, grad, flow, boun, ctrl, iter, time, hflx, hist, cont, rlp, rlpm, FEHM stress module, CO2 module, and trac. 3. Reading, modification and writing of FEHM restart files. 4. Running FEHM simulations including directory management, construction of control files. 5. Post-processing of output data, including slice plots, profile plots, time series, 3-D cutaway. 6. FEHM thermodynamic calculator. Calculates density, viscosity and enthalpy of liquid and gaseous water for given temperature and pressure. Provides same information for CO2 if FEHM's CO2 interpolation table is available. 7. Limited support for template models. At present, a 2-D wellbore model and quarter-domain EGS problem are included. 8. Miscellaneous tools: conversion between lat, long and UTM coordinates. PyFEHM is based on a similar set of Python libraries for the LBNL simulator TOUGH2, called PyTOUGH. PyTOUGH is released under the Gnu Lesser General Public License (same as intended for PyFEHM).	8/21/2013

13-082	C13152	Cesium GoogleEarth Enterprise Imagery Provider, Version 1.x	The Cesium GoogleEarth Enterprise Imagery Provider is a Javascript library that extends the Cesium ImageryProvider Javascript Interface. This library connects to a GoogleEarth Enterprise server 2D globe and will serve the requested tiled imagery based upon the requested X and Y coordinates, and Zoom level. The imagery must be in either a flat" geometric projection or "Mercator" projection.	8/21/2013
13-083	C13154	SIMD Kernels for Divergent Execution Paths, Version 1.0	SKDEP is a set of computational kernels for algorithms in which data parallelism is inhibited by divergent execution paths. The kernels allow experimentation with a variety of treatments to resolve execution path divergence and related problems, including coalesced memory reads and writes. SKDEP includes implementation code for several architectures such as traditional CPU and GPGPUs. Kernels are intended to be reimplemented in other languages and programming models as a means of evaluating computational strategies on various platforms.	8/22/2013
13-084	C13155	PyVXI-11, Version 1.0	PyVXI-11 is a Python extension to interface with electronics laboratory equipment such as oscilloscopes, network and spectrum analyzes, multimeters, etc., using the standard SCPI language and TCP/IP. Python is a perfect language for manipulating SCPI, which is text, and the numerical results produced by electronics laboratory equipment, using NumPy. This package is a modern pythonic alternative to LabView, Linux GPIB, and VISA. PyVXI-11 supports service request interrupts (SRQ), asynchronous I/O, and other advanced requirements for automating an electronics R&D laboratory.	9/4/2013
13-086	C13164	yelpy, Version 1.x	yelpy is a simple, small, yet complete, pure Python implementation of the Yelp 2.0 API (http://www.yelp.com/developers/documentation/v2/overview).	9/5/2013
13-087	C13156	CoulMD	The software is a molecular dynamics code specifically designed to simulate plasmas. The code solves the classical equations of motion for a given set of particles interacting through the Coulomb force. The code is based on a parallel implementation of the particle-particle particle-mesh (PPPM) algorithm, which combines high resolution for individual encounters with a rapid, mesh-based, long-range force calculation. The code include subroutines to calculate physical properties of plasmas, including equilibrium and transport properties.	9/4/2013
13-088	C13163	1D Hydro Code	This one-dimensional Lagrangian hydrocode was written to simulate gas dynamics. It has the capability to run different simulations using a variety of discretization methods. These discretization methods include cell-centered, staggered-grid, and point-centered methods. A first order Riemann solver is implemented in the code as well as a fourth order Runge-Kutta time integration scheme. The purpose of this code's development was to provide tools to explore new methods and algorithms in computational hydrodynamics.	9/5/2013
13-089	C13162	GIANT (GUI-based Image ANnotation Tool), Version 1.x	GIANT (GUI-based Image ANnotation Tool) is a software tool used to annotate TIFF (Tagged Image File Format) imagery. GIANT was written using the Interactive Data Language (IDL; see exelisvis.com). GIANT provides an easy-to-use graphical user interface (GUI) that allows a user to load an image and immediately begin to create point-based labels which are displayed as an overlay on the image. The annotations are derived from a default (or customized) Object Controlled Vocabulary (OCV). Annotation changes (additions or deletions) are continuously recorded to a GIANT Annotation File (GAF), so that saving or reloading annotation information occurs transparently to the user. GIANT provides image zoom, pan/scroll and rotation capabilities, allowing extensive image manipulations during an annotation session. It also continuously updates an image-to-display transformation. The inverse of this transformation is used to convert user-selected annotation locations from display coordinates to image coordinates, so that annotation locations are properly recorded in image coordinates to the GAF file. GIANT handles both grayscale and color TIFF image files. It has only been used on orthoimagery produced from aerial survey imagery, but it should be applicable for use with any TIFF imagery.	9/5/2013
13-091	C13161	Explosive Shock Image Analysis System (ESIAS), Version 1.x	Explosive Shock Image Analysis System (ESIAS) is designed to perform quantitative analyses of fast-frame imagery taken during the detonation of high explosive (HE) rate stick shot experiments. HE rate sticks are used to provide a consistent measure of HE behavior in order to tune the parameters of equation of state models so that the HE behavior can be simulated by hydrodynamic models. ESAIS can be used to extract velocity measurements from the shock and product gases from aquarium shots and the metal cylinder expansion from cylinder shot experiments. Due to the fact that light refracts at the shock surface due to a difference in densities between the shocked and undisturbed water, it is necessary to correct for the resulting magnification effect on any surface observed through the shock surface. ESIAS employs ray-tracing techniques to correct for this magnification so that an accurate measure of these surfaces (and subsequently the calculated velocities) can be obtained. ESIAS also includes the capability to load and analyze data captured from other instruments during the HE detonation in order to facilitate the comparison, synchronization, and analysis of the various data streams.	9/5/2013

13-092	C13160	ProcVision, Version 0.x	ProcVision is a malware detection system that is designed to identify evidence of exploitation (i.e., Indicators of Compromise) within volatile memory. The system employs a combination of signatures, heuristics and machine-learning classification to identify suspicious activity and malicious processes in both user and kernel memory space. Not only can it be used to detect malware and exploitation attempts in real-time, but it can also be used in a forensics capacity to discover the extent of exploitation of known vulnerabilities. A simple plugin framework allows for rapid testing and deployment of new capabilities and integration with other information systems.	9/5/2013
13-093	C13159	CodeVision, Version 0.x	CodeVision (formerly MassAV) is a malware analysis workflow system and repository developed at LANL. It began as a simple tool for analyzing a large malware corpus using multiple COTS antivirus engines in parallel. It has since evolved into a unified interface for collecting, storing and analyzing malicious executable code using a variety of COTS and GOTS backends. Not only can CodeVision receive samples from users via its command-line and web interfaces, but it can also operate on continuous network data feeds (e.g., perimeter e-mail attachments, HTTP downloads). The modular analysis framework and simple API give developers the ability to integrate custom and off-the-shelf analysis tools with ease. The system manages scheduling and dependencies, allowing modules to be chained together into analysis-driven workflows. The system is designed as a distributed, load-balanced web service, an architecture that allows it to scale horizontally by adding additional nodes. The CodeVision system has been in continuous use at LANL since 2008 and has become a staple tool for the both computer incident response team and malware analysis researchers. CodeVision achieves excellent detection accuracy (more than 98%) by combining the results of many different analysis methods into a single determination of maliciousness. The system employs advanced machine learning and statistical methods to classify executables based upon a variety of static and dynamic features, including instruction-level execution traces. Unlike signature-based methods, these techniques have the potential to detect zero-day malware. The method for combining multiple data views in CodeVision not only increases overall classification accuracy, but significantly increases accuracy when the user desires to set a low false positive rate. This makes CodeVision much more attractive when being deployed on live networks.	9/5/2013
13-094	C13157	script name: restart_entrust_launch_agents.py Version 1.0	When installing the Java Crypto Extension(JCE), we need to restart the entrust client(with this script). Restarting entrust services is required, so the entrust services will re-read the libraries. Without the JCE, users are not able to use the entrust AES 256 encryption feature. This occurs with Java 1.7. We need to distribute the JCE and use this script to restart the entrust services. Currently the way we are restarting of the services seems to disable the interface so it is not usable until after a restart of the machine. We would like to be able to restart the services without restarting the machine, or requiring people to log out and log back in for the services to be started correctly.	9/5/2013
13-095	C13165	Empire, Version 1.x	The Empire software is a refactorization of a legacy code called CHAP which was written here at LANL in the late 1960s. The purpose of the code is to compute the electric field resulting from a gamma ray burst in the upper part of the earth's atmosphere. The code reads the user's input choices and checks them for errors. If the input deck is correct, the gamma source data is read from a file (if necessary) and the gamma energy is stored in the appropriate data structures. Next, a temporal and spatial mesh is generated according to the instructions from the input deck. This serves as the computational grid for the simulation. Next, electrons are injected into the system and their momenta are computed. From the momentum calculations, the current densities are computed to be used later by the Maxwell Equation solver. An air chemistry computation is also performed to get a value for the air conductivity and electron number density at each mesh point for each time in the calculation. The conductivity is also used in the Maxwell solver. Finally, the code prints out the electric and magnetic fields at each point in the spatial mesh for at all time steps. This data is displayed in columns to make plotting the results easy.	9/13/2013
13-096	C13169	DOPRI5.f90	DR_DOPRI5_new.f90 is the driver for DOPRI5.f90. This code is based off of the Method of Lines code developed by Dormand and Prince. This is an explicit Runge-Kutta code with stepsize control that solves a system of first order ordinary differential equations. This code solves the electron Boltzmann Equation which is finite differenced in energy space for increasing time steps. It also solves for the elastic and inelastic collision frequencies of these electrons in the electric field specified. This will be used for my research for LANL in electron transport physics.	9/18/2013

13-097	C13168	Signet, Version 1.0	Signet is a flow oriented signal processing framework that distributes arbitrary signal processing calculations to threads running on a multi-core computer. Typically, one would write a set of modules subclassed from signet::node in C++ or Python and use the prescribed Signet API, that each perform some part of a larger calculation and use Signet to handle message passing, parallel distribution across your computer, scheduling, and coordinating Monte Carlo trials or replicants. Some general purpose nodes are included in the base distribution, such as HDF5 I/O and Fast Fourier Transforms built on top of FFTW. The Signet engine in this base distribution contains no science or specialized calculations; these must be written by the user or acquired from a separate repository. The author has used Signet to integrate much open source software such as FFTW (Fourier analysis), Meep (FDTD electromagnetics), HDF5 (standard scientific file format), ffmpeg (video encoding and decoding), gdal (image encoding and decoding) all into one multithreaded application. Custom modules have been previously used for particle filters, multi-taper spectral analysis, computational neuroscience, electronics laboratory automation, and many other signal processing algorithms.	9/18/2013
13-098	C13167	LANL-OKMC 1.x	The software, herein referred to as OKMC, is an object kinetic Monte Carlo simulator that is capable of incorporating the effects of strain. OKMC is capable of computing diffusion paths of point defects and defect clusters in arbitrarily strained environments. OKMC allows for reactions between diffusing defects, and interactions of defects with other more complex micro structural features (such as dislocations, voids, etc.). OKMC uses modern C++ constructs like vectors for efficiency and readability of the code.	9/18/2013
13-099	C13170	HOSS (Hybrid Optimization Software Suite)	The hybrid optimization software suite (HOSS) is a general purpose combined finite discrete element (FDEM) code with an integrated solid-fluid solver technology that also includes customized based applications for the mining industry (MCAPS = Mining Customized Analyst Production Software) and oil and gas industries (OGCAPS = Oil and Gas Customized Analyst Production Software). MCAPS and OGCAPS will be Java-driven utility packages that will be customized to solved a set of real-world problems required by practitioners in both mining and oil and gas industries.	9/18/2013
13-100	C13171	MUNROU-ISF	The MUNROU-ISF is a general purpose combined finite discrete element (FDEM) code with an integrated solid-fluid solver technology. The code has been designed to handle solid fracture and fragmentation modeling problems using a combination of the finite element method and discrete element method. The code is fully parallel for 2D and 3D operations, allows full implementation of an extensive array of plasticity material models, and utilizes state of the art contact and fracture algorithms. In addition, a pseudo-Eulerian fluid solver has been incorporated into the code in order to take into account the effect of the presence of a fluid medium inside the fractures and/or inside the solid matrix. Finally, an enhanced visualizer is incorporated to allow processing of multi-millions of elements.	9/18/2013
13-101	C13166	WAAS (Warnings And Alerts Service), Version 4.x	The launch Warning and Alerts Service (WAAS) will support analysts and operators in detecting threats and answering other important questions related to launch such as launch type as well as generating alert messages for approval and release. The threat detection capability is provided by the LANL Integrated Knowledge Engine (IKE), a mature, web-centric Bayesian analysis tool. WAAS integrates manual, semi automated and automated data feeds and is being developed for threats and other important questions related to Launch. The service alerts operators to potential problems, supports the operator in drilling down into the details that caused the alert, and also supports planning of further data to be collected.	9/17/2013
13-102	C13172	tsk_get_files.py, version 1.0	The open source project "The Sleuth Kit" (TSK) is a software package for recovering data lost on a computer. It has two specific tools - "fls" and "icat". "fls" is for listing files on a drive or drive image. "icat" can recover a single file, based on the information from "fls". This software wraps those two functions to recover multiple files, based on a search string, that would be a directory, or part of a file name.	9/18/2013

13-103	C13173	stiff.f90	This code is designed to analyze the evolution of a swarm of electrons that is created by photoelectron ionization of air and Townsend impact ionization at a given air density. The code uses an adaptive timestep, which is calculated based on an error between test solutions of high and low order, and solves a system of coupled differential equations for electric field, electron temperature, swarm electron number density, and drift velocity given in Higgins, Longmire, and O’Dell (1973). This is done by using a variant of a 4th order Runge-Kutta method called the Rosenbrock method. This method is used to solve a stiff set of equations and is an embedded approach.	9/18/2013
13-104	C13174	Sequedex, Version 1.0	Sequedex is computer software to rapidly assign phylogeny and function to short sequences of DNA. It also identifies protein-coding regions of DNA, characterizes complexity of a data set and similarity to other data sets. It also characterizes DNA sequencer performance.	9/26/2013
13-105	C13175	Sequescan, Version 1.0	Sequedex is computer software to rapidly assign phylogeny and function to short sequences of DNA. It also identifies protein-coding regions of DNA, characterizes complexity of a data set and similarity to other data sets. It also characterizes DNA sequencer performance.	9/26/2013
13-106	C13176	Tree of Life Genomic Signatures Database, 2055-node Version_public	Database of genomics signatures for use with the Sequedex software package. This database associates DNA signature sequences with phylogenetic and functional identifications. The entire Tree of Life (excluding viruses) is covered by this database, including bacteria, archaea, and eukaryotes.	9/26/2013
13-107	C13178	Tree of Life Genomic Signatures Database, 2055-node Version	Database of genomics signatures for use with the Sequedex software package. This database associates DNA signature sequences with phylogenetic and functional identifications. The entire Tree of Life (excluding viruses) is covered by this database, including bacteria, archaea, and eukaryotes.	9/26/2013
13-108	C13177	Multi Infrastructure Control and Optimization Toolkit (MICOT), Version 1.x	MICOT is a tool for optimizing and controlling infrastructure systems. In includes modules for optimizing the operations of an infrastructure structure (for example optimal dispatch), designing infrastructure systems, restoring infrastructures systems, preparing for natural disasters, interdicting networks, state estimation, sensor placement, and simulation of infrastructure systems. It implements algorithms developed at LANL that have been published in the academic community.	9/26/2013