1. Abstract
Motivated by the microprocessor industry transition from single to many-core chip designs, this project investigates a framework that leverages heterogeneous computing to tackle many-body dynamics problems. The heterogeneous computing infrastructure relies on a combination of Central Processing Units (CPUs) and Graphics Processing Units (GPUs); the latter regarded as co-processors or accelerators, capable of delivering substantial efficiency gains in simulating many-body dynamics problems. Examples of such problems include granular material dynamics, simulation of fluid-solid interaction, and molecular dynamics analysis.

3. A Heterogeneous CPU/GPU HPC Environment for Computational Dynamics Applications
Computational dynamics aims at determining how a system of mutually interacting elements changes in time. Heterogeneous computing, that is, parallel computing drawing on both CPU and GPU hardware, becomes relevant when the number of interacting elements is very large. High-performance heterogeneous computational dynamics requires:
(a) the ability to partition the problem according to a one-to-one mapping (i.e., spatial subdivision: one subdomain to one computing unit);
(b) a protocol for passing data between any two computing units;
(c) algorithms for element proximity computation;
(d) the ability to carry out post-processing for data analysis and visualization in a distributed fashion.

4. Computational Dynamics: Many-Body Dynamics Analysis
When handling rigid and flexible bodies interacting through contact and friction, the physics-based system simulation requires the solution of a combined set of differential-algebraic equations and inequality constraints:

The goal is to devise numerical solution procedures for the above problem that map well onto heterogeneous hardware: i.e., that leverage both CPU and GPU computing.

7. Computational Dynamics: Fluid-Solid Interaction Analysis
Fluid-Solid Interaction Analysis relies on the Smoothed Particle Hydrodynamics (SPH) method. In this Lagrangian method, each particle carries field variables (density, internal energy, velocity, etc.). A kernel-function approach defines the influence area of each particle. Field variables and their derivatives can be approximated as outlined in Fig. 8. The approach is embarrassingly parallel since each computational thread can independently process one particle. The original partial differential problem is reduced to an ordinary differential problem by expressing the right-hand side using SPH approximations for field function derivatives and an appropriate state equation.

8. Technology Demonstration: Fluid Sloshing in Moving Tanker
A classic example of fluid structure interaction is the tank fluid sloshing problem (Fig. 9). Approximately 230,000 points define the tanker boundary geometry and 300,000 SPH particles discretize the liquid. After the fluid particles have settled, the tank is set into a motion that mimics a typical driving maneuver and the fluid starts to slosh around in the tank. Figure 10 reports the sloshing forces in three directions, that act on the tanker. Gravity acts in negative y-direction. The tanker starts to accelerate in positive z-direction and drives through curves in positive and negative x-direction. The simulation was performed in approximately 19 hours using a NVIDIA GeForce 8800 GTX graphics card with 128 Scalar Processors.

9. Conclusion
The key observation that motivates this effort is that commodity hardware available today has tremendous compute power. What is missing is the solution methods and software that harnesses these heterogeneous supercomputers. The long-term expected outcome of this project is a software infrastructure that can be used by any researcher interested in using heterogeneous CPU/GPU hardware for HPC in Computational Dynamics applications. Preliminary results obtained with an early version of this software infrastructure show one to two orders of magnitude reductions in simulation times for real-life dynamics applications.