Principles of Software Engineering: Course Outline

Ethan Jackson And Wolfram Schulte,
Research in Software Engineering (RiSE)
Microsoft Research

Overview

- Motivation and Focus
- Syllabus
- Projects

i. Motivation and Focus

http://www.cs.washington.edu/csep503

Motivation

Software engineering is more important than ever, from many points-of-view:

- **Business point-of-view**: Need to sell software with few bugs in the face of ever shorter release cycles. Badly engineered software is counter-productive.

- **Consumer point-of-view**: Software should perform its functions quickly, correctly, securely, privately, using little power… and the list continues to grow.

- **Societal point-of-view**: Software helps to drive our cars, monitor our health, generate our power. How can we engineer software that lives up to these applications?

A Journey…

This class will be a journey through state-of-the-art techniques in software engineering.

Focus: Engineering Correct Software

Our journey has a focus.
Engineering correct software that is:

- Likely to be correct, for some properties, by extensive automated testing.
- Provably correct, for some properties, by automated deduction.
- Provably correct, for some properties, by automated synthesis.

Breadth: Try Many Techniques…

Our journey has twists and turns.
We don’t know of a single technique that address all SE problems:

- **Programming in the Small:**
  How do we design a solitary algorithm without bugs?

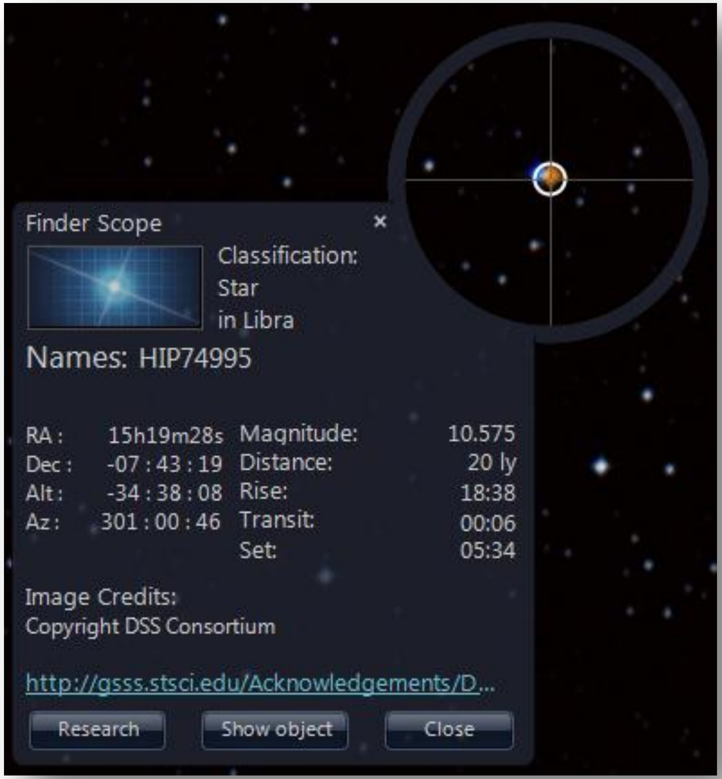
- **Programming in the Large:**
  How do we orchestrate several concurrent software systems?

- **Programming in the Real:**
  How do we write software that controls a physical system?

Mission: Get to Gliese 581

Our journey has a mission.

Design a probe that travels to the nearest earth-size planet in habitable zone.



Probe must autonomously travel 20 light years, which translates to 8,645 bug-free years at 10 times fastest speed ever achieved.

ii. Syllabus

http://www.cs.washington.edu/csep503

Part 1: Programming in the Small

First two weeks focus on functionally correct sequential software modules using pre/post conditions (code contracts):

- **Class 1**: Static verification using code contracts and abstract interpretation. Experiment with the tool Clousot.

- **Class 2**: Full static verification using object/loop invariants and automated theorem proving. Experiment with the Dafny language.

Part 2: Programming in the Large

Second two weeks focus on orchestrating concurrent systems using (timed) automata theory.

- **Class 3**: Automata-theoretic models - Untimed and timed automata. Build models of system orchestration using the tools SMV and Uppaal.

- **Class 4**: Specifying temporal properties with LTL, CTL, CTL*, and observers. Model checking to verify temporal properties.

Part 3: Programming in the Real

Third two weeks look at synthesizing systems from models.

- **Class 5**: Realizing system synthesis through code generation. Applying formal verification to code generators.

- **Class 6**: Design-space exploration of software/hardware architectures in the presence of resource constraints. Experiment for the FORMULA system.

Part 4: Simulating Systems

Fourth two weeks look at simulation before implementation.

- **Class 7**: Simulation of mixed-domain models. Problems of combining discrete/continuous systems. Experiment with Ptolemy II framework.

- **Class 8**: Low-cost prototyping by simulation of virtual hardware. Experiment with the Giano system.

Part 5: Empirical Software Engineering

Fifth two weeks look at empirical software engineering.

- **Class 9**: Approaches to bug predication. Applying bug data from other related projects to predict bugs new projects.
- **Class 10**: Using data analytics to make decision. Impact of organizational structure on bugs.

Projects, Grading, and Homework (I)

- **Tests**: None.

- **Projects**: Class is project based. There will be 4 two-week projects. Leaves one week of start-up time, and one week of slack in case more time is needed.

- **Groups**: Feel free to work in groups of two.

Projects, Grading, and Homework (II)

- **Time**: Expect to spend several hours to: (1) get a new tool up-and-running, (2) think through the problem, (3) solve the problem. Probably 6 – 8 hours a week is reasonable.
- **Grading**: Historically, this class focuses on the journey. A strong attempt at projects guarantees a high grade.
- **TA \ Labs**: There is no TA and not set lab times. The class is small enough that we can meet in a lab if that is helpful.

Projects, Grading, and Homework (III)

Most importantly, have Fun!

iii. Projects

http://www.cs.washington.edu/csep503

Background (I)

In 1995 the first extra-solar planet was definitely confirmed.

http://www.nature.com/nature/journal/v378/n6555/abs/37

A Jupiter-mass companion ...

nature

Thursday 05 January 2012

article

Nature **378**, 355 - 359 (23 November 1995); doi:10.1038/378355a0

A Jupiter-mass companion to a solar-type star

MICHEL MAYOR & DIDIER QUELOZ

Geneva Observatory, 51 Chemin des Maillettes, CH-1290 Sauverny, Switzerland

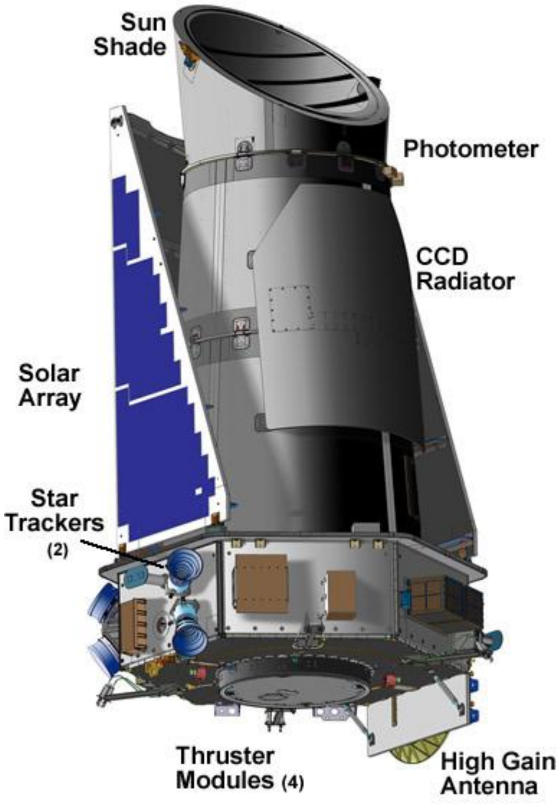
The presence of a Jupiter-mass companion to the star 51 Pegasi is inferred from observations of periodic variations in the star's radial velocity. The companion lies only about eight million kilometres from the star, which would be well inside the orbit of Mercury in our Solar System. This object might be a gas-giant planet that has migrated to this location through orbital evolution, or from the radiative stripping of a brown dwarf.

References

1. Walker, G. A. H., Walker, A. R. & Irwin, A. W., *Icarus* **116**, 359–375 (1995). | Article |
2. Cochran, W. D. & Hatzes, A. P. *Astrophys. Space Sci.* **212**, 281–291 (1994). | Article |
3. Marcy, G. W. & Butler, R. P. *Publ. astr. Soc. Pacif.* **104**, 270–277 (1992).
4. McMillan, R. S., Moore, T. L., Perry, M. L. & Smith, P. H. *Astrophys. Space Sci.* **212**, 271–280 (1994). | Article |
5. Marcy, G. W. & Butler, R. P. in *The Bottom of the Main Sequence and Beyond* (ESO Astrophys. Symp.) (ed. Tinney, C. G.) 98–108 (Springer, Berlin, 1995).

Background (II)

In 2009 the Kepler mission was launched to look for planets among 100,000 stars.



To date Kepler has identified over 2,326 planets.

Background (III)

Dec. 20th, 2011 Kepler finds two earth-size planets.
Jan. 1, 2012 the BBC predicts a habitable earth discovered within the year.


http://www.bbc.co.uk/news/science-envirc

BBC News - A

BBC Mobile News | Sp

NEWS SCIENCE & ENVIRONME

Home | US & Canada | Latin America | UK | Africa | Asia | Europe | Mid-East | Busin

1 January 2012 Last updated at 05:30 ET

612

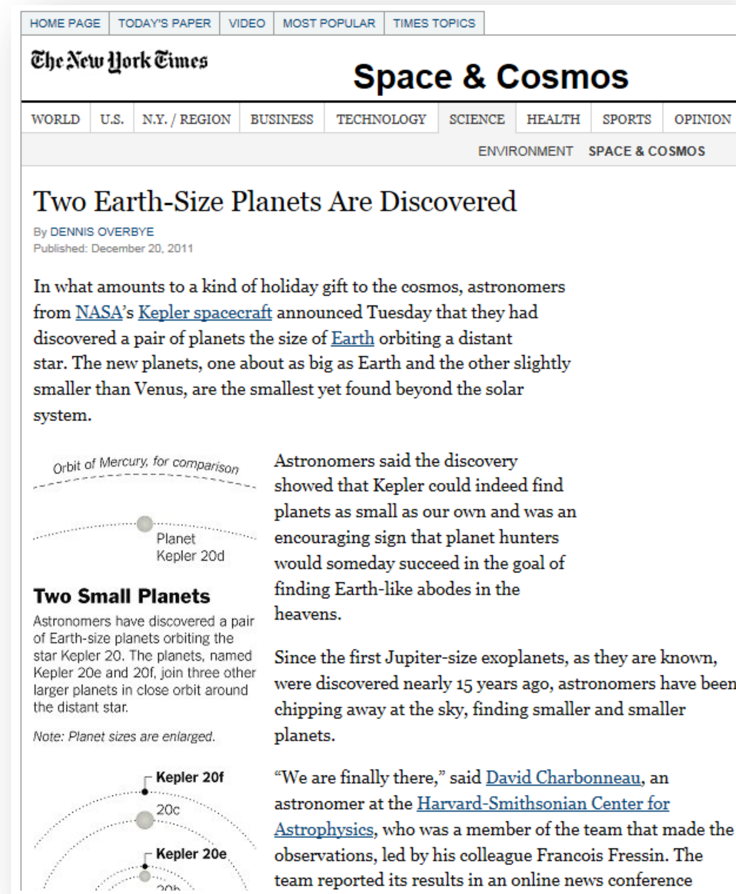
A science news preview of 2012

By Jason Palmer and Paul Rincon
Science & Environment desk, BBC News

A home from home

The discovery of planets orbiting other stars has become so commonplace that each new one barely elicits mention in the media (**unless they come as a six pack**). The Kepler space telescope is consistently turning up exoplanet candidates in bulk; at the time of writing it had found a total of 2,326 of them (but there are probably more by the time of reading). What seems inevitable is that one of them will end up being a dead ringer for Earth.

Already telescopes have confirmed a Kepler find as **the first roughly Earth-sized planet around a Sun-like star that could play host to water** - but the confirmation of a far-flung rocky world with oceans and an atmosphere awaits. There are 47 promising chances in Kepler's catalogue already, and plenty more yet to be confirmed. Next year we are likely to see something decisively more like our home planet, and we can then move on to the more daunting but more intriguing questions about whether other planets host life.


HOME PAGE | TODAY'S PAPER | VIDEO | MOST POPULAR | TIMES TOPICS

The New York Times

Space & Cosmos

WORLD | U.S. | N.Y. / REGION | BUSINESS | TECHNOLOGY | SCIENCE | HEALTH | SPORTS | OPINION

ENVIRONMENT | SPACE & COSMOS

Two Earth-Size Planets Are Discovered

By DENNIS OVERBYE
Published: December 20, 2011

In what amounts to a kind of holiday gift to the cosmos, astronomers from NASA's Kepler spacecraft announced Tuesday that they had discovered a pair of planets the size of Earth orbiting a distant star. The new planets, one about as big as Earth and the other slightly smaller than Venus, are the smallest yet found beyond the solar system.

Two Small Planets

Astronomers have discovered a pair of Earth-size planets orbiting the star Kepler 20. The planets, named Kepler 20e and 20f, join three other larger planets in close orbit around the distant star.

Note: Planet sizes are enlarged.

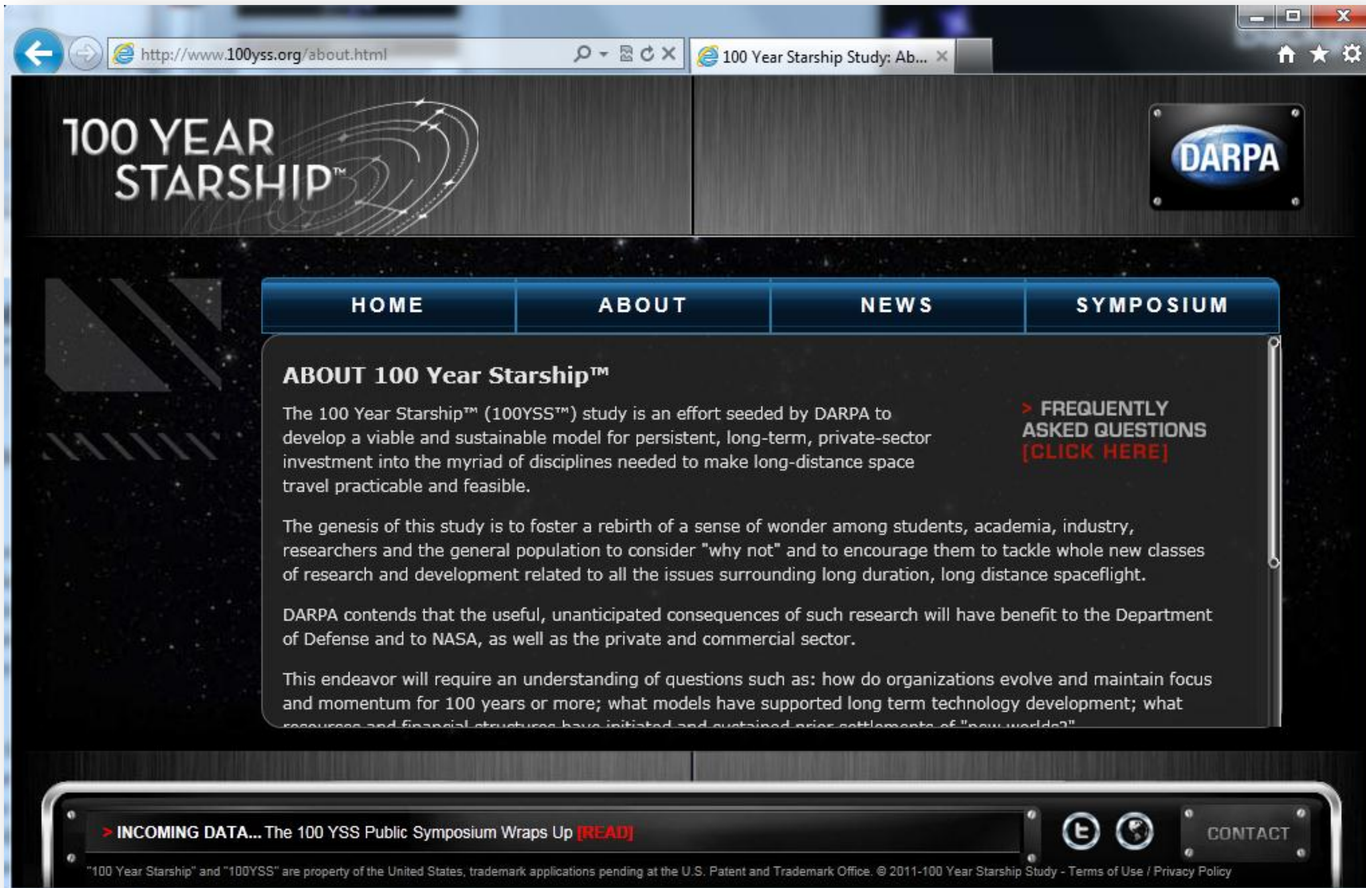
Astronomers said the discovery showed that Kepler could indeed find planets as small as our own and was an encouraging sign that planet hunters would someday succeed in the goal of finding Earth-like abodes in the heavens.

Since the first Jupiter-size exoplanets, as they are known, were discovered nearly 15 years ago, astronomers have been chipping away at the sky, finding smaller and smaller planets.

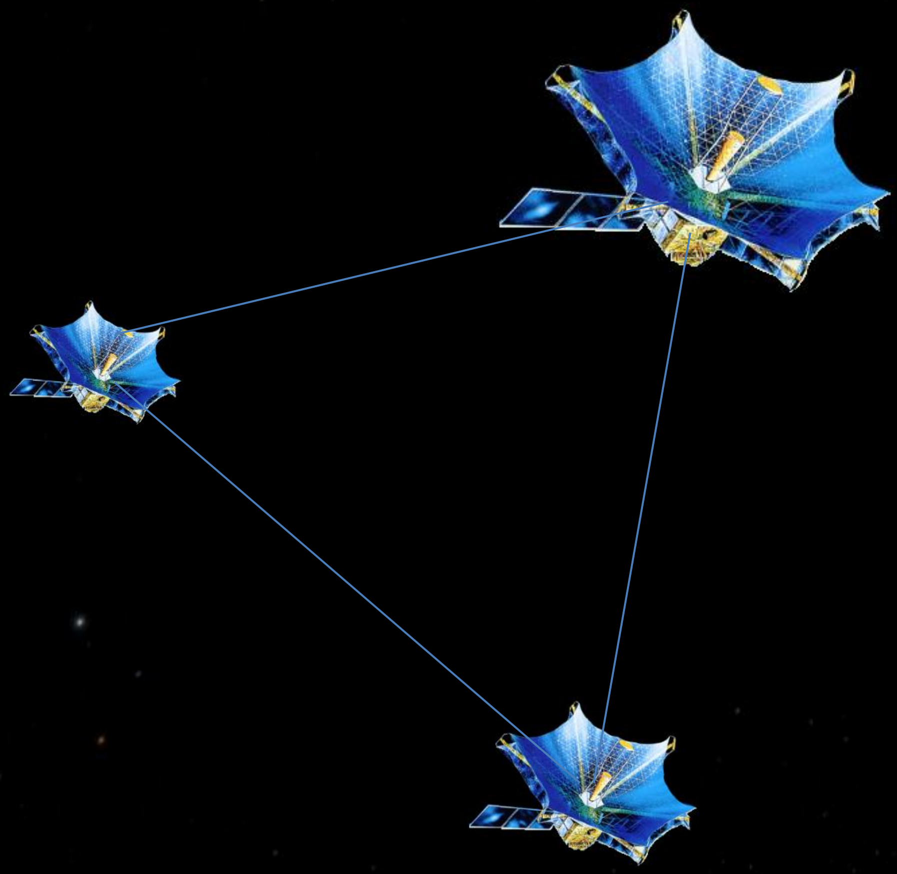
"We are finally there," said David Charbonneau, an astronomer at the Harvard-Smithsonian Center for Astrophysics, who was a member of the team that made the observations, led by his colleague Francois Fressin. The team reported its results in an online news conference

Background (IV)

Jan. 5th, 2012 (today) DARPA leaks that astronaut Mae Jemison will head the "100 Year Starship" project to develop a starship.



Probe 503



Project 1: Astrometrics Subsystem

Probes must determine their location in space without any help from earth.

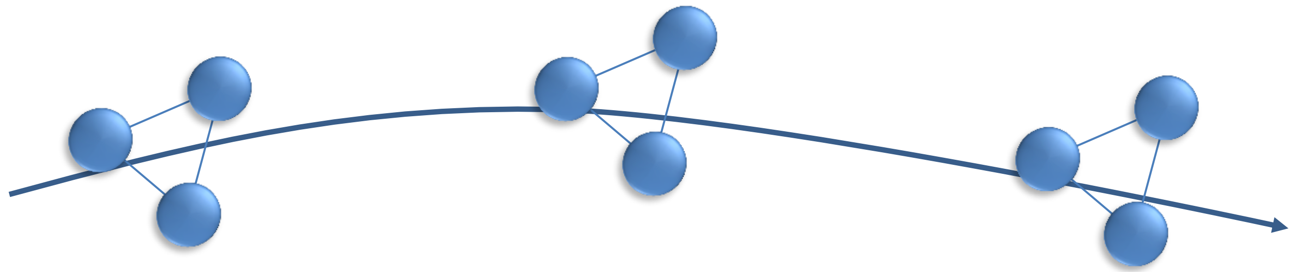
- **Location in ICRF**: Probe must find its direction in the ICRF by matching observed radio-sources with a database of known radio-sources.

- **Design and Prove**: Write a subsystem that matches a region of the sky with an database of known markers to determine space craft orientation. Prove it correct.

Project 2: Command-and-Control

Three probes must move in tandem separated by 1 AU. Devise a command-and-control system that preserves this requirement.

- **Command-and-control**: Model command-and-control system as a set of interacting timed automata. These handle the high-level operations of the probe (e.g. call the astrometrics subsystem).

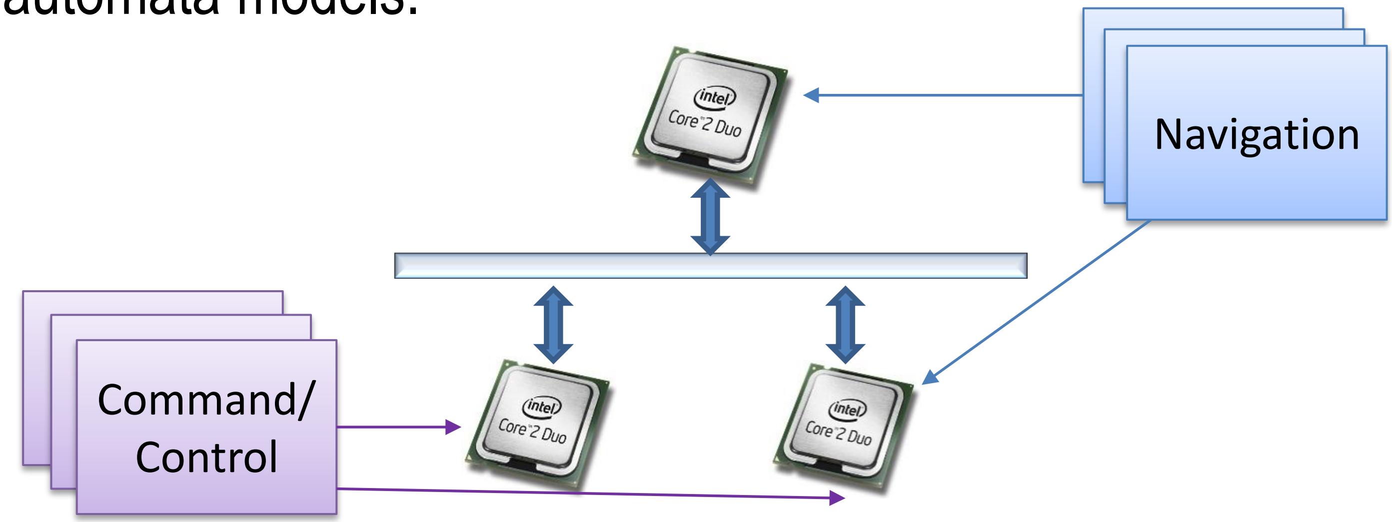


- **Prove**: Specify temporal properties of the command-and-control system. Use Uppaal explicit state model checking to verify properties.

Project 3: Synthesize Probe System

Generate software and partition onto hardware.

- **Code generation**: Write a code generator that produces an implementation of the command-and-control-system from automata models.

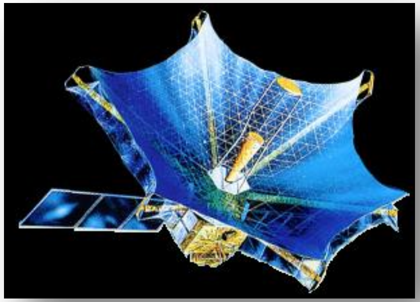


- **System synthesis**: Specify software/hardware partitioning problem as a constraint system over resources and synthesize candidate architectures.

Project 4: System Simulation

Simulate behavior of synthesized system using Ptolemy II

- **Plant Model**: Build a simple model of probe dynamics using continuous-time models in Ptolemy II.



- **Hybrid Model**: Combine plant model with synthesized discrete-time system to simulate complete behavior of probes.

Project 0: Play with Code Contracts

1. Get Visual Studio up and running. CSE students can obtain it for free:
   http://www.cs.washington.edu/lab/sw/MSDNAA/ms-sw.html
   (Just need Profession version.)

2. Get Code Contracts at:
   http://msdn.microsoft.com/en-us/devlabs/dd491992

3. Read the documentation and try some of the samples.

Thanks And Questions!

http://www.cs.washington.edu/csep503