

Usability study of Amphion

Description

Amphion/NAIF is a program synthesis system that creates programs to solve problems in the domain of solar system mechanics. It is available over the World Wide Web from a server at NASA Ames Research Center.

This report documents the findings of a usability study of Amphion/NAIF. The study was conducted at Adams State College in the first quarter of 1999. The intent of this study was to determine if Amphion's user interface is usable by high school science and mathematics instructors and students and to identify how to improve the user interface for Amphion. One of the anticipated outcomes of this study is a set of suggestions for student projects or lesson plans which utilize Amphion/NAIF.

A small set of student volunteers at ASC were given access to Amphion and the Amphion tutorials. The students were juniors and seniors majoring in either computer science or mathematics/mathematics secondary education. These students were observed as they worked through the tutorial and were given small problems to solve. Two of the students were given a one hour introduction to Amphion and then worked through the tutorial. The other two students were given only a brief introduction (five minutes) and were asked to complete the tutorial. The following sections analyze the observations made during the students efforts.

I have made every effort to faithfully represent the comments and actions of the subjects.

General comments from the users about the system:

The students found Amphion interesting, specifically that the data, programs, and simulations were based on "real" NASA data and that it is possible to create programs from simple diagrams. The students who were given an introduction to Amphion found it easier to use than those who did not. For the most part, those given an introduction skipped most of the preliminary writing in the tutorial; those without the introduction read through the preliminary "motivation" paper, and felt confused more than enlightened by it.

The students found it easy to create simple specifications, but tedious to create more complex specifications. This was apparently due to (1) a lack of full understanding of astronomic concepts such as light time corrections; (2) an inability to visualize the 3-dimensional constructs (such as the angle between two planes being the same as the angle between their normal vectors); (3) unfamiliarity with the objects that can be created (for example, knowing that a point can be created by projecting a point onto a plane makes solving the solar eclipse specification easier); and (4) unfamiliarity with the terminology

(ephemeris time, J2000.). When students had difficulty with terminology, there was very little help available.

In general it takes a significant amount of knowledge to create a useful program. These students may not have had an adequate background to solve the problems. The students with programming background fared better than those without. The problem solving skills developed by creating programs seems to translate to solving problems using Amphion. On a few occasions, these students were frustrated by the lack of ability to visually compartmentalize or divide a problem.

At least one student suggested that from the first tutorial, it was not clear that it was easier to use Amphion to solve hard problems.

Things users liked

- The pulldown menus work once the user becomes familiar enough with them. These do in fact lead the user in the creation of objects.
- The first tutorial seems to be at the right level, but needs to be less text and more substance. Needs better hyperlinking for new material. The basic mode was to look at the pictures, try to do operations, read only enough to get them through.

Things users didn't like

- The random placement of new objects is annoying.
- Difficult to figure out what the meaning of an arrow is.
- Not very helpful in identifying problems. Example of overconstrained spec.
- Not very helpful in manipulating things on the screen. Take the overconstrained spec and modify it by removing some things. Lots of seemingly reasonable operations not allowed.

Things that seemed to be difficult for users

- Several places in the tutorial make references to items with which the users were not familiar. For example, geodetic coordinates (what is a flattening factor and what values work here)?
- Several instances of disallowed operations when it seems like they should be allowed. Example: create a specification with Earth, Sun, Mars. Specify time of all three planets to be the same, an input time. Then connect the three with photons. This is an unsatisfiable spec. The problem is the times are overconstrained. What we want is to delete the arcs connecting the input time from the definition of Sun and Mars. This operation is not allowed.
- No indication about reserved words
- The concept of body is difficult. They invariably initially confused body and body-id. The concept of a planet at a given time is not difficult, and once they were assisted, this seemed clear. Initially, it's not obvious what the difference between the two is.

Recommendations for the tutorial

- Put more hyperlinks into the tutorials.
- Cut the amount of text,
- give pointers to things that are not general knowledge.
- include a glossary of terms?
- The tutorials should tell the user that Amphion is not case-sensitive.

Recommendations for Amphion in general

- Remove the unusable parts from the Add menu: since there are no “define new operations”, gray this out or remove it.
- Put in a more useful information feature. The File Information (?) doesn’t give a user anything useful. What would be nice is a description of what operation was used to create this thing.
- Have a searchable help page that describes the objects and when they can be used.
- Make a color scheme that selects objects of a certain type and colors them (to assist reading).
- Make a "reorder" command that places objects in a reasonable spread on the screen. Shake, scramble, don't do anything useful
- fewer steps to create objects.
- Search seems to be an operation with a high likelihood of making it into lesson plans
- enhanced visualization: for example: view the solar eclipse problem from some vantage point that makes it obvious that the moon does not orbit in the ecliptic plane.
- Remove or disable unusable parts
- provide some lay description of objects in specs
- make spec comments searchable

Possible lesson plans

Intro to programming: seems to incorporate elements of top-down design

search for eclipse

search for occultation

size of moon on horizon: actual vs computed (compare atmospheric effects)

compute kepler ellipses from observation data

Bugs

There is no response whatsoever when an unsatisfiable specification is submitted.

Notes from observations
Specific comments on aspects from users

Logan Hansen:

No indication that there are reserved words. Why can't a body be named "earth"

Kelly

3/3/99

Started tutorial at 8:07. Started reading motivation. This part of tutorial is the SISN paper from 1994. It's long and tedious. This should be summarized, hyperlinked, and streamlined. It should be possible to get through this in a few minutes, with very little detail about how to build things. She spent no time looking at the Fortran code or the specification. "The only thing that's cool is the description of the name." Finished at 8:17.

It was a little bit confusing for her to have things on the screen that look exactly like windows. She clicked on the netscape screen objects several times instead of the spec she was building.

Task 1.4 8:22

This part of the tutorial repeats much of the information presented in the motivation. This is the right place for most of it, closer to the actual use. The font is pretty small. Somehow we need fewer words, more actions, and perhaps push some of the verbiage to hyperlinked pages?

Crashed amphion at 8:27.

Randy:

Would like delete key support

Attempted phases of the moon problem. Found keyboard access lacking. Could not figure out how to do a projection. The problem is in visualizing the scenario. No simple way to build a visualization and view from different perspectives.

Needs search: search for alignment, search for eclipse, search for max and min.

Kelly Aulds

The paper on motivation is a terrible way to start the tutorial. It's way too long, too hard to read on line, does not motivate much.

It's really not obvious to "click here to start".
It's not obvious to add an email on the first time.

This user tends not to read long sections of text.

Things run slowly. Nearly a minute nothing happened after login. User found this confusing.

Something should alert the user that the title bar changes when the spec is renamed.

This user had to read 4.1 several times to get the point.

BUG: can't drag the body icon down from the top of the window.
The ERROR message in add-body with same name is not very helpful or forgiving.

Cullen Stevens:

Shake and scramble don't help at all. These should be removed from the tutorial.
The color layout doesn't seem logical.

This user skipped most of the reading.

Declare I/O is never used: should be removed?

FORTTRAN variable discussion in the tutorial is off the mark. It seems to be at the wrong level and distracts from creating a spec.

Got "comma not inside of backquote" error on a drag and drop into spec comment window.

So many windows pop up that it is difficult and confusing to find which are new, which are on the tutorial page and which are from amphion. Need to clean this up some.
Maybe make the background different

Tutorial description doesn't have seconds.

Something is wrong with full names for inputs and outputs.

Thought the graphical layout could correspond to a program design

March

Week 1	4 hr kelly
Week 2	2 hr phil/
Week 3	no time
Week 4	2 h r outline of usability

Apr

Week 1	4 hr cullen
Week 2	4 hr specs and stuff w/ ttp
Week 3	4 hr suggestions of fixes/ usability study
Week 4	