N94-23902

Making Tomorrow's Mistakes Today: Evolutionary Prototyping for Risk Reduction and Shorter Development Time

by Gary Friedman, Ursula Schwuttke, Scott Burliegh, Sanguan Chow, Randy Parlier, Lorrine Lee, Henry Castro, and Jim Gersbach

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drvie
Pasadena, CA 91109

Introduction

In the early days of JPL's solar system exploration, each spacecraft mission required its own dedicated data system with all software applications written in the mainframe's native assembly language. Although these early telemetry processing systems were a triumph of engineering in their day, since that time the computer industry has advanced to the point where it is now advantageous to replace these systems with more modern technology.

The Space Flight Operations Center (SFOC) Prototype group was established in 1985 as a workstation and software laboratory. The charter of the lab was to determine if it was possible to construct a multi-mission telemetry processing system using commercial, off-the-shelf computers that communicated via networks. The staff of the lab mirrored that of a typical skunk works operation -- a small, multi-disciplinary team with a great deal of autonomy that could get complex tasks done quickly.

Although today many computer manufacturers are committed to converging on software standards and are scurrying to provide a common interface, seven years ago this wasn't the case. UNIX System III was ruled by AT&T, Sun was not a major player in the then-new workstation market, and TCP/IP was predicted to be dead by the Department of Defense in less than two years.

In an effort to determine which approaches would be useful, the prototype group experimented with all types of operating systems, inter-process communication mechanisms, network protocols, packet size parameters. Out of that pioneering work came the confidence that a multi-mission telemetry processing system could be built using high-level languages running in a heterogeneous, networked workstation environment. Experience revealed that the operating systems on all nodes should be similar (i.e., all VMS or all PC-DOS or all UNIX), and that a unique Data Transport Subsystem tool needed to be built to address the incompatibilities of network standards, byte ordering, and socket buffering.

The advantages of building a telemetry processing system based on emerging industry standards were numerous: by employing these standards, we would no longer be locked into a single vendor. When new technology came to market which offered ten times the performance at one eighth the cost, it would be possible to attach the new machine to the network, re-compile the application code, and run. In addition, we would no longer be plagued with lack of manufacturer support when we encountered obscure bugs. And maybe, hopefully, the eternal elusive goal of software portability across different vendors' platforms would finally be available.

The Prototype Group (later renamed the Flight Projects Office Information Systems Testbed, or FIST) performed a evolutionary prototype of this new architecture, uncovered potential problems, and learned a great deal about multi-vendor interoperability. All of this heavily influenced the eventual implementation of the world's first distributed multi-mission telemetry processing system.

Looking back on that experience, one marvels at the risky decisions that were made to standardize on UNIX, TCP/IP and C. These decisions have since been validated by UNIX vendors who are all clamoring to move to "Open Systems" in order to strengthen their market share.

Since that time, FIST has been responsible for keeping JPL's flight projects abreast of emerging technologies, such as user interface and data visualization tools, networking, artificial intelligence, distributed systems, databases, and other promising software technologies in order to determine how these might benefit both the mission operations and scientific communities. We showed Voyager scientists real-time graphical data during the Neptune encounter. We've demonstrated the feasibility of graphical planning tools and shown mission operations teams how today's tools can be applied to make their day-to-day activities more manageable. But most importantly, FIST has been a resource that can help JPL remain cutting edge with minimal cost and risk.

Some highlights of our prototyping efforts are described in more detail later in this paper

The Advantages of Evolutionary Prototyping

Evolutionary prototyping refers to high-quality programs used to validate or uncover requirements, and to validate a possible design. Unlike a throw-away prototype, which is useful when many of the aspects of a design are untried, evolutionary prototypes are robust in design and are built upon a foundation of that which is well-understood. Evolutionary prototypes are designed to be built upon, incorporating foresight and software hooks to allow for expansion and eventual delivery.

Why do evolutionary prototyping? Three primary reasons: risk reduction, shortened development cycle, and accelerated technology transfer.

The idea of pre-development evolutionary prototyping has already proven its value to many JPL projects. It is a fast and cost-effective method of proving out new concepts and accelerating their simultaneous integration into operational environments and next-generation products.

Prototyping can hasten the transfer of new technology into large systems. The short-loop feedback cycle from operational settings and user communities results in real user requirements being dynamically defined. The result is a useful product in a shorter time and without the inherent risk of untried technologies. Evolutionary prototypes require less documentation, comply with limited formal specifications, and involve informal configuration management and fewer formal verification tests.

In addition, this type of prototyping concludes in the worst case with a small-scale, limited distribution product for operational environments, and in the best case leads to full-scale development of multi-user and multi-mission systems.

Highlights of FIST Prototypes and Pilots

VNESSA

To support Voyager II's historic encounter with Neptune, the FIST lab prototyped a new science visualization system called VNESSA (Voyager Neptune Encounter Science Support Activity). This pilot project encompassed both the user interface as well as the telemetry processing engine needed to drive it, and was the first time any of FIST's prototypes presented a system view rather than individual pieces of the telemetry processing puzzle.

VNESSA represented several "firsts" for Voyager. This was the first time that fields and particles investigators had access to their data electronically in near-realtime. Recently acquired near-realtime data were also stored in a database for the first time, and could be recalled later. It was also the first time that this data could be displayed in multiple, animated color graphical "windows" in a computer screen under the control of the science user. For the first time, VNESSA allowed these investigators to provide graphical illustrations of their discoveries to the news media almost immediately after receiving the near-realtime data. There was even a speech synthesizer connected to the operator's console which allowed operators to audibly monitor critical events even when they were elsewhere in the support area.

The science teams used VNESSA more heavily during the encounter than was expected, and the system's usefulness exceeded the expectations of most participants. The VNESSA-developed data displays were unexpectedly popular for correlating results from two or more instruments, verifying the science

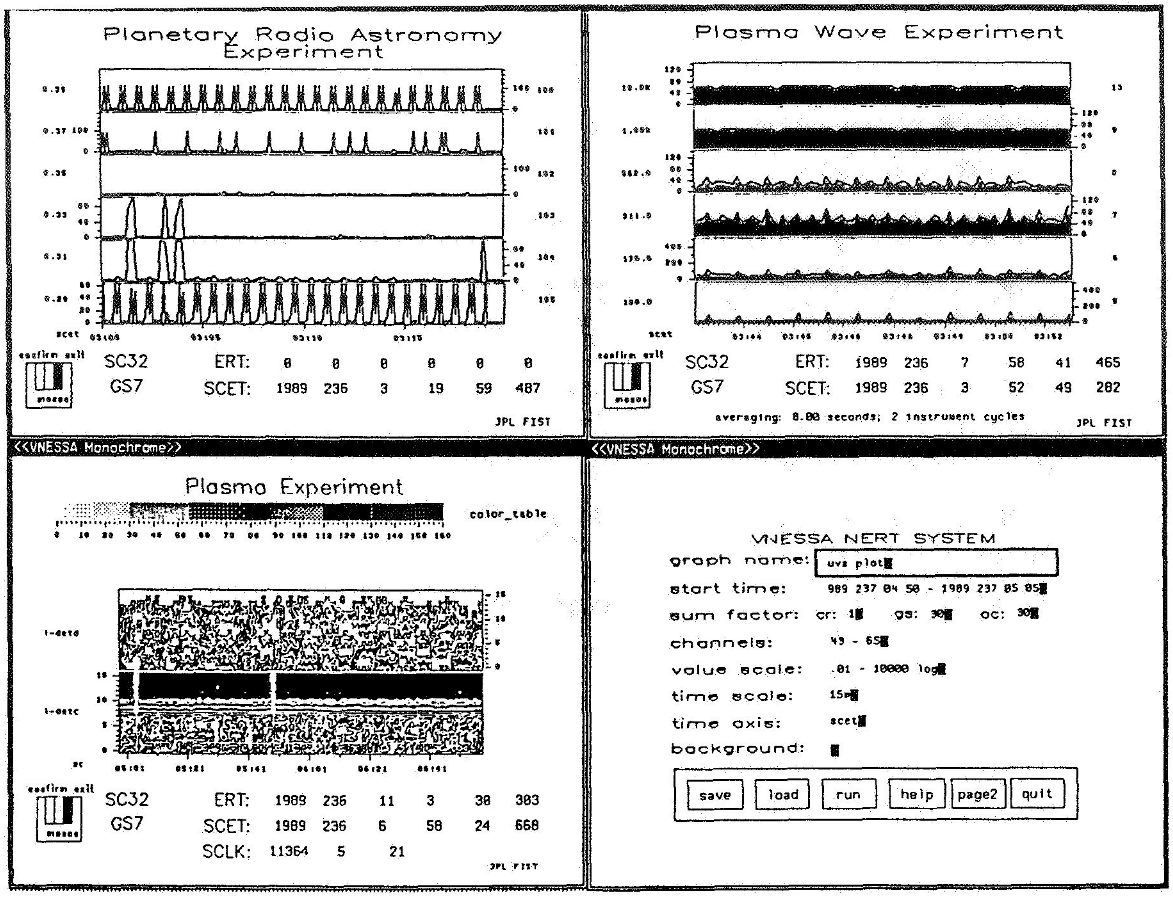

Figure 1. *VNESSA provided Voyager Principal Investigators with access to Neptune encounter science data in near real time and provided enhanced near real time science data processing capabilities. Three VNESSA displays for the Neptune encounter show plots for Planetary Radio Astronmy (PRA) and Plasma experiments.*

teams' own analyses, and supporting members of teams whose own equipment could not accommodate any more users. We also accidentally discovered that putting all the Principal Investigators and their near-real-time data displays into one room can catalyze scientific discovery: one team's discovery of a milestone event can confirm and enhance the conclusions of the other teams.

Lessons Learned from VNESSA

This prototype taught us a lot about building real-time high-volume telemetry data servers on top of commercial database products which were designed for more conventional use. Considerable energy was devoted to developing work-arounds to these database problems, most of which were caused by unusual Instrument Data Frame lengths and volumes. Other lessons learned fell into the hindsight category: although we started 18 months early and had regular dialog with the science teams, FIST archives proclaim that we should have started sooner, should have had more interaction among participants, should have placed VNESSA workstation displays in the Voyager General Science Data Team's work area.

Overall, VNESSA reinforced the notion that, using commercial tools and de facto industry standards, a small, skilled, and stable development team can perform impressive feats and rapidly usher in new technology much faster than normally would have occurred.

SANTA

To support the Galileo spacecraft's first two major events, FIST also performed evolutionary prototyping to create a UNIX-based telemetry processing system which worked in parallel with the original Galileo data stream. The name SANTA (Science Analysis Near-Term Activity) was chosen since both activities - the initial 4-day instrument checkout and the "Earth-1 encounter" (the first time Galileo passed near the Earth on its gravity-assisted trajectory to Jupiter) - occurred near Christmas time in their respective years.

The first Science Analysis Near-Term Activity

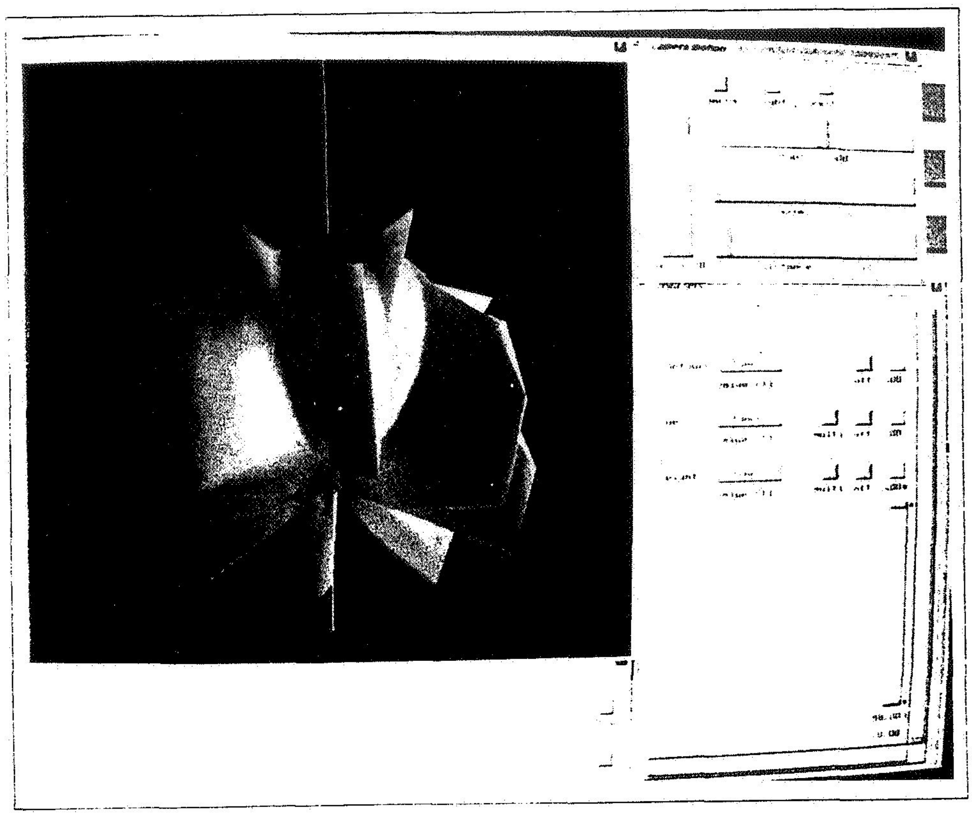

Fig. 2 ***SANTA II was prototyped to support Galileo's Earth-1 encounter. It delivered low-rate science in near real time to geographically dispersed Principal Investigators in a secure fashion over the Internet. In addition, many new visualization schemes were tried. Above is a prototype "3-D" phase-space plot of plasma particles. Each picture represents a particle type sampled using one revolution of the spacecraft's spun bus.***

(SANTA) was a demonstration of the quick-look science support component planned for the Space Flight Operations Center. To accomplish this goal, the VNESSA Instrument Data Frame server was adapted to process Galileo data. It accurately delivered low-rate science data to Galileo's scientists in real time (generally about 6 seconds after receipt at Earth) for four days; during this time it dropped only two minutes of recoverable data.

The Galileo scientists were reportedly pleased with the system and very interested in using this kind of real-time science data access extensively in the future, both for the upcoming Galileo encounters and for other space flight missions.

SANTA II (December, 1990) was built for the Galileo Earth-I encounter, and strove to do for Galileo what VNESSA had done for Voyager: show the project the kinds of things possible when using the SFOC architecture. The project successfully brought together talent from three different JPL sections, half a million dollars' worth of loaned computer equipment, and a pre-alpha version of a newly-redesigned SFOC.

SANTA II had a wide range of aspirations, among them prototyping of new and "equivalent" science displays, demonstrating new SFOC subsystems, trying out new computer technology, and the secure distribution of data via the Internet. Like VNESSA, we had near-real-time science displays whose parameters were user-definable.

The area of largest impact was the controversy over security procedures. Users wanted seamless, remote, automated access to our telemetry data server, while management, concerned about external system break-ins and viruses, insisted on some mechanism to ensure access only to valid users. A brain-dead UNIX box (with all the net.daemons turned off and only one active network port) was used in conjunction with challenge-response authentication tokens, which required the users' presence and precluded any automation of the data gathering process.

Lessons Learned from SANTA I and II

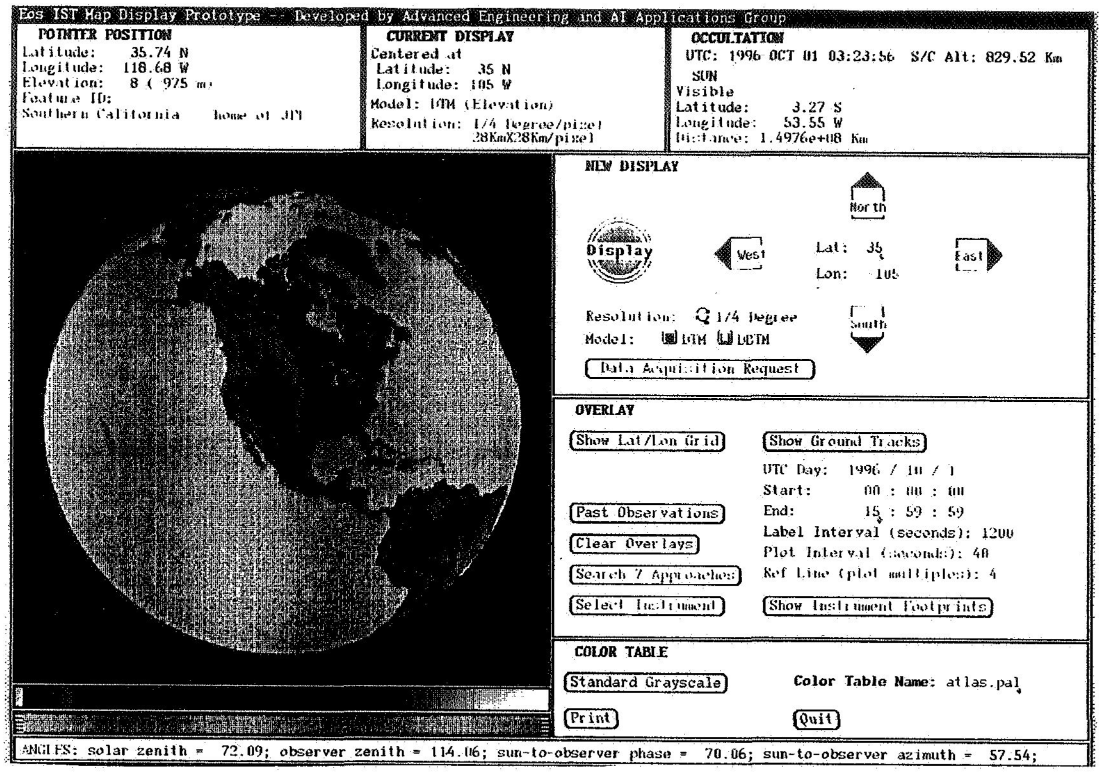

Fig. 3 *The EOS Instrument Support Terminal (IST) prototype can be used to visualize spacecraft ground tracks during the planning process. It originally sought to clarify, refine, and discover requirements for the operational IST. In the illustration above, the spacecraft's path is superimposed over the current view of the planet, showing position and orientation for anygiven time.*

The security experiment taught us the dangers of being relaxed about security in order not to offend users. Although the External User Access Node's design was sound, our choice of a small PIN size as a user password for the authentication tokens eventually encouraged development of an illicit program that mimicked the token's behavior. No aspect of the system was compromised; but we could no longer be sure active sessions were started by authorized individuals. (Since that experiment, the EUA concept has been replaced by Kerberos and a dedicated router with manually-updated tables, so only trusted hosts from fixed locations can receive packets.)

Some of the SANTA II innovative displays brought praise from the science teams, and are now being improved upon for Galileo's upcoming Earth II encounter. Other lessons from the SANTAs have been incorporated into the SFOC Telemetry Delivery Service (TDS) and have helped improve the stability of the SFOC Galileo adaptation.

MOPSIE

The object of the Mars Observer Pilot SOPC Implementation Effort (MOPSIE) was to prototype a suite of software, integrated under some sort of graphical user interface, that runs on workstations called SOPCs (Science Operations Planning Computer). The SOPCs are used by planetary exploration mission scientists for several purposes, including monitoring the health and status of the instruments on the MO spacecraft, planning and issuing instrument command sequences based on the computed position of the spacecraft at various times, and exchanging messages and data among themselves and with the SFOC subsystems.

MOPSIE gave us an idea of what effect relatively slow long-distance network communications will have on remote operations. Lessons learned from MOPSIE are especially important as we move from an era of "fly-by" missions to one of long-term orbital missions; scientists (understandably) don't want to remain at JPL throughout the data acquisition phase of an orbiter's mission.

Future Visions

FIST is continuing to keep abreast of new technologies and prototyping their application with an eye on immediate needs. Among the projects we have lined up in the near future:

- Sequence Verification: Developing a computer model (State Tracker prototype) of a spacecraft's behavior can help in the planning and debugging of command sequences.

- Small Mission Prototype: Numerous smaller missions are likely to become more prevalent in JPL's future and are likely to replace flagship missions like Voyager, Galileo, Cassini. To support the current NASA vision of small missions to each of the planets, FIST is exploring missions operations systems tailored to this new concept.

Our first application of such a system is MESUR (Mars Environmental SURvey) Pathfinder, a mission designed to explore issues relevant to success of a future network mission which aims to place 16 small landers, some or all accompanied by micro-rovers, on the Martian surface.

The MESUR project is considering a wide range of innovations in system development that might cut costs. One such innovation is early integration of the end-to-end data system -- that is, development of flight software in an environment that includes the existing Multimission Ground Data System, to encourage early communication of design decisions among flight and ground software developers.

FIST may be the prototype for the Pathfinder's System Development Laboratory, which will enhance communication among developer by the establishment and use of a single development facility.

- VULCAN (Visualization Utility for Locating Coronal Accelerated Nucleons), a Solar Flare visualization tool employing 3-D modeling software packages to help convert mountains of raw data into scientific understanding.

- Prototypes of small distributed systems built upon OSF's Distributed Computing Environment (DCE), enabling us to announce whether these promising multi-vendor services are mature enough to be built upon. Evaluations are being done with other industry standards (OSF/1, POSIX-compliant system calls, new workstations, and laptop UNIX boxes.

Conclusion

FIST's task of being a "technology gatekeeper" continues to ease new technology into current and future projects within JPL. Having such a prototyping front-end can reduce both risk and development time, while simultaneously ensuring that the products incorporate worthwhile new ideas. Evolutionary prototyping can play a significant role in achieving NASA's new goals of smaller, faster, and cheaper space missions.

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.