IT middleware services for an "Exploration Web"

SMC-IT Conference 2003
S18: Future Software Directions

Norm Lamarra

Funded by JPL-IND Technology Program
16 July 2003

Overview

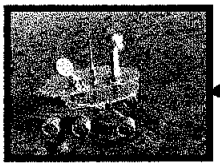
- Background & Problem Domain
  - Space Communications Context
  - Some Assumed Problem Areas
  - Proposed Solution Approach
  - Key Common Concepts
  - Conceptual Evolving Shared IT Services
  - Object Messaging (FY02 work)
- FY03 focus: Remote Science Data Management
  - Example Science Applications: Current/Future
  - Approach & Goals for Demo
  - Demo...
- Conclusions
  - Issues Addressed by this Demo
  - Summary Benefits of Service Approach
  - Layered Service View
  - Directions for Future Work

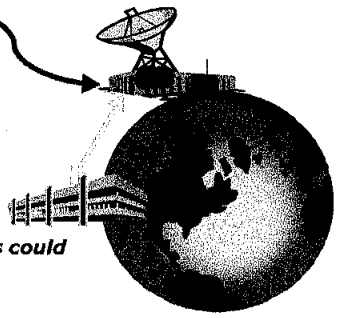
Space Middleware Context

Remote sensors and other resources could be used more effectively...

Space Middleware Services

- Metadata Service
- Comms Service
- Persistence Service
- Pub/Sub Service





Science access could be simplified...

N Lamarra 3

Some Assumed Problem Areas

- Loose Scientist-Instrument Connection
  - Coordinating mission plans is cumbersome and costly
  - Coordinating spacecraft is difficult (constellation or formation)
- Insufficient Data Value
  - Sensors increasing resolution, but data delivery limitations
    - Which bits to d/l (MGS <1%)? Data vs. Information vs. Knowledge
  - Difficult to perform coordinated measurements
- High Operations cost
  - Automation can reduce cost (e.g., automated data management)
  - Autonomy can increase capability and reduce cost
    - On-board reasoning (e.g., vehicle health, science goals, etc.)
- Difficult Application Development
  - Few standard API's or accepted s/w architecture
  - Difficult to coordinate among distributed resources
  - Limited robustness (e.g., failure det'n/recovery, s/w modification)

Note: use increased on-board storage/processing to reduce req'd link b/w...

N Lamarra 4

Proposed Solution Approach

- Conceptualize a set of standardized "*shared services*"
  - 3 broad categories: Communications, Storage, Processing
  - Distributed client-server model useful for all 3
    - Make object model highly flexible
    - Make clients as lightweight as possible
    - Simplify server replication (when necessary)
  - Build upon "enhanced" internet-style communication
    - Asynchronous messaging has many advantages
    - Publish/subscribe has further advantages
    - Message prioritization and efficiency are crucial
- Deploy "*layered infrastructure*" incrementally
  - Basic services: Messaging, time, events, security
  - Information services: data management, alarms
  - Higher-level services: navigation, weather, etc.
  - Agent interaction infrastructure (far future)
    - e.g., "autonomous" communication vs. "scheduled"

Time

N Lamarra 5

Key Common concepts

- Extensible Shared Object Model (e.g., SharedNet)
  - Client works with local objects (vehicle, sensor, etc.)
    - Create, modify events distributed via publish/subscribe
  - Server maintains current value (or history) for distribution
  - Only attributes and object references are transferred (efficient)
    - Objects can be "meta-objects" or even "meta-meta-objects"
- Higher-layer information processes
  - Constructed from lower-layer data
    - events, values, locations, etc.
- Common applications:
  - Monitoring (Sensors, Data Sources, Services, ...)
  - Information fusion (Situation awareness, ...)
  - Analysis (Distributed, automated, ...)
  - Decision support (Agent assistance, ...)

N Lamarra 6

Prototype Shared IT Services

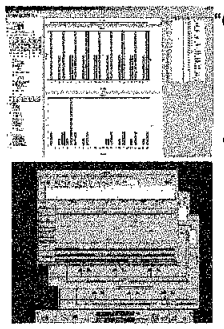
- **Communications**
  - Tolerate delay, disconnection, b/w limitation
    - Buffered, asynchronous, ...
  - Tolerate variety of network topologies (near/far) *FY02*
    - Simplify data relay
  - Provide QoS (guarantees, reserved b/w, etc.)
    - Allow (dynamic) priorities (inc. time-to-live)
  - Allow choice of transport protocol
    - Support standards (e.g., CCSDS)
- Processing (on-board & distributed)
  - Simplify science processing
  - Support fault tolerance (service management)
  - Simplify off-board processing (like "solver service") *FY03*
- Storage
  - Provide flexible storage type (e.g., image, meas't, stream)
  - Provide query/retrieval capability
  - Support management functions (e.g., location, access)
  - Simplify transport (e.g., move, replicate)

N Lamarra 7

FY02 Prototype (Object Messaging)

Example Scenario	Shows

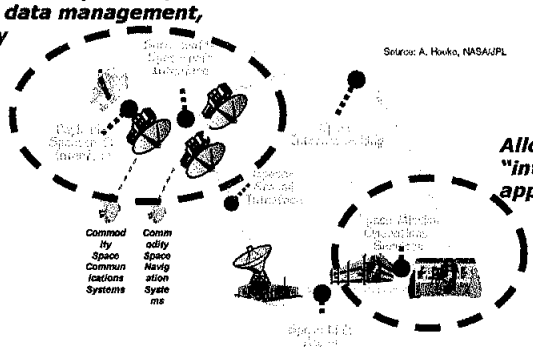
• Remote Planning Coordination:
  • e.g., MER-A/MER-B/ODY
• Plan change by one affects others
  • Time criticality (view periods)
• Negotiations reach a solution
  • Minimal use of link to Earth
	• "Ad-hoc" remote comms
	  – Robust MOM: buffered, async, QoS,...
	  – Extensible message object model
	• "Subscription" by message type
	  – Simple client (Java API, C++ wrapper)
	• "GUI client" displays filtered traffic
	  – Can join "after the fact"



N Lamarra 8

FY03 Prototype (Science Data Mgt)

Enable remote applications to simplify science acquisition, processing, data management, and delivery



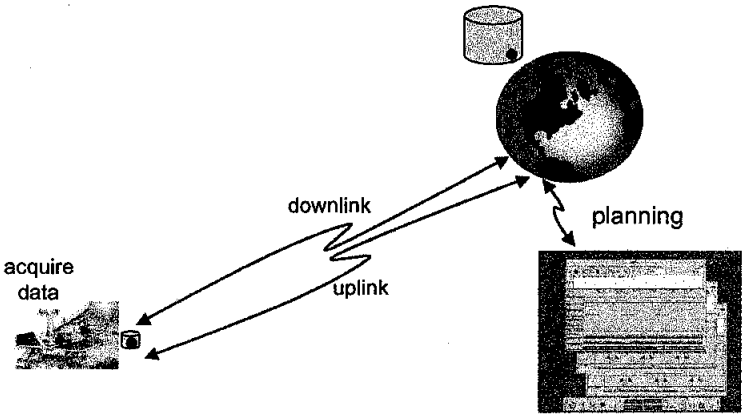
Allow science user to "interact" with remote applications and data

Remote sensors and other resources could be used more effectively...

Science access could be simplified...

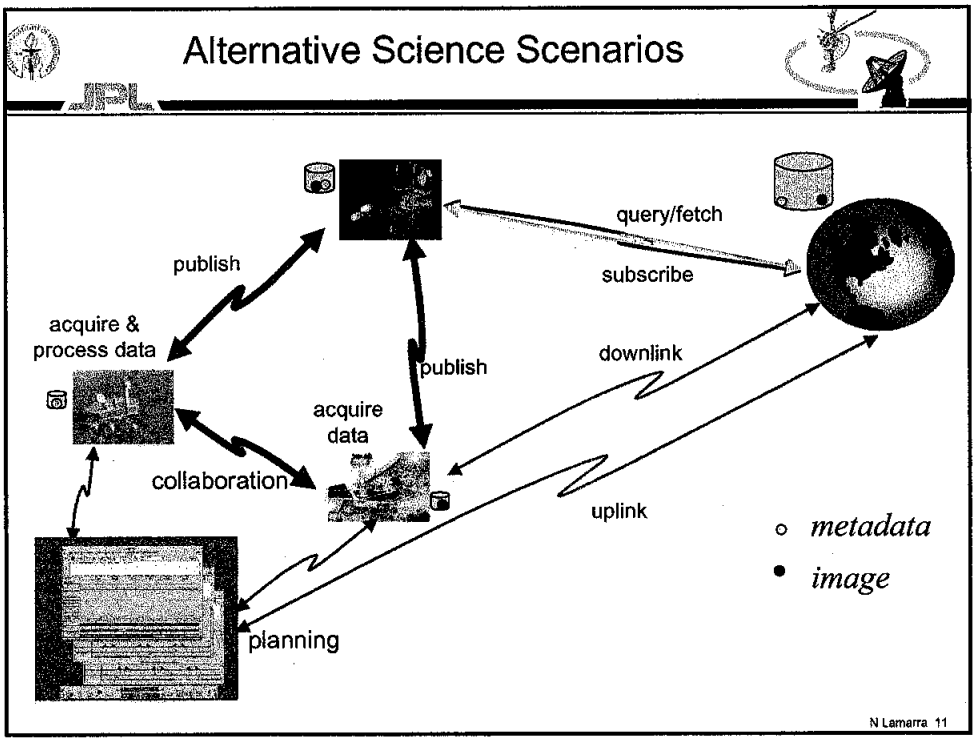
N Lamarra 9

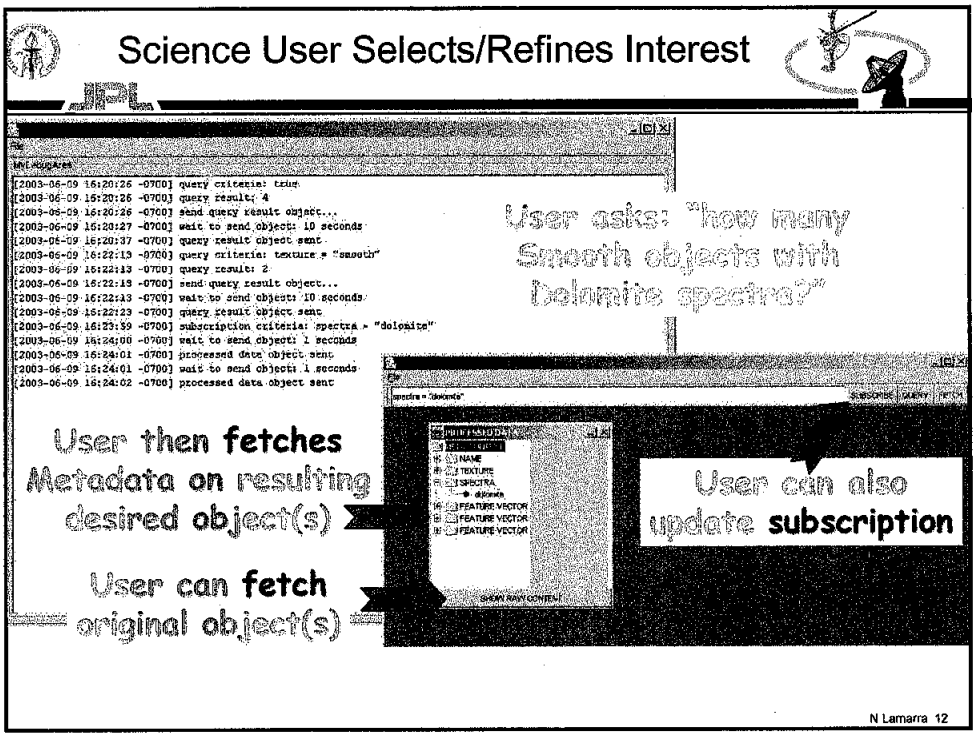
Simple Current Science Scenario



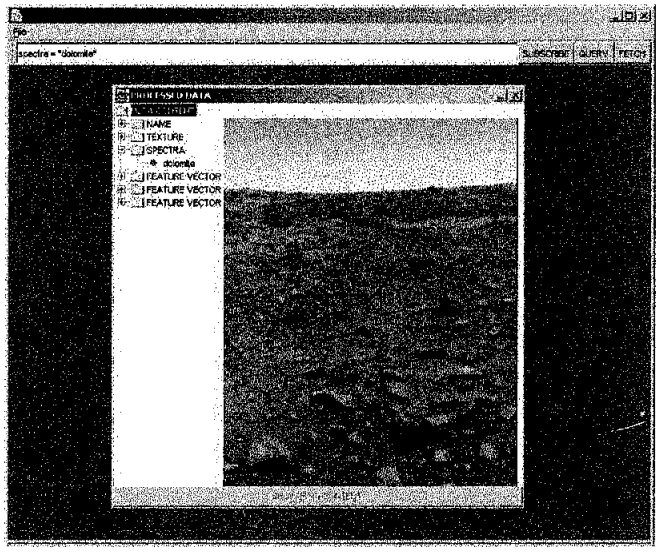
- *image*

N Lamarra 10


Alternative Science Scenarios
JPL
query/fetch
publish
subscribe
acquire &
process data
publish
downlink
acquire
data
collaboration
uplink
metadata
image
planning
N Lamarra 11


Science User Selects/Refines Interest
JPL
User asks: "how many Smooth objects with Dolomite spectra?"
User then fetches Metadata on resulting desired object(s)
User can fetch original object(s)
User can also update subscription
N Lamarra 12

User Fetches Desired Image



Conclusions

"Middleware-enabled Exploration" Issues

- Efficient use of bandwidth
  - Uplink/downlink
  - Usefulness of metadata
- Automated use of remote resources
  - Storage, communication, processing
- Potential to facilitate adaptive scheduling
  - Interface to on-board or off-board planner
    - Including schedule for infrastructure itself (e.g., comm)
- Improved quality of sensor data
  - Efficient handling of bandwidth constraints
  - Flexible remote data manipulation (even remote data mining)
- Generalized framework for distributed applications
  - Leverages robust infrastructure

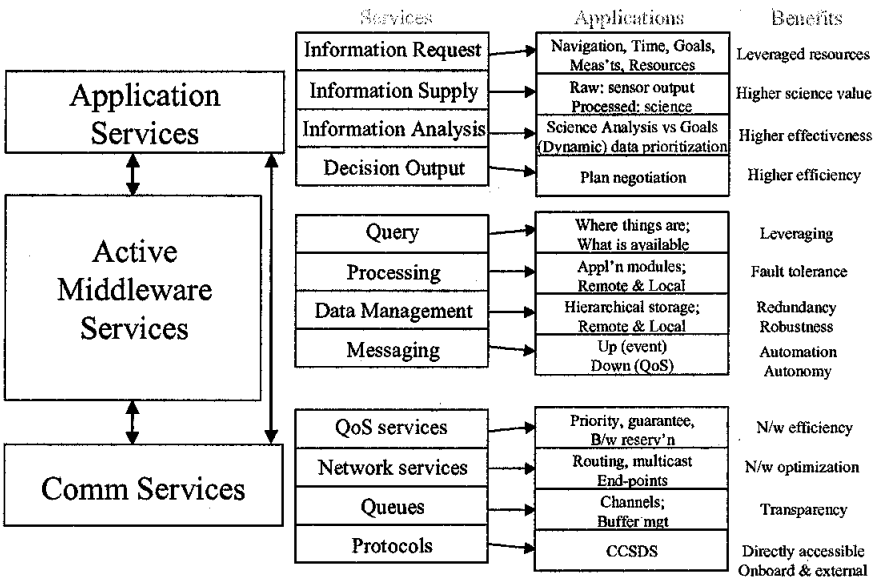
N Lamarra 15

Summary Benefits of Service Approach

- Simplified communications
  - Improve use of "local" network bandwidth
    - Higher aggregate capacity than typically "scheduled" for use
    - Lower latency, redundant, fault-tolerant, adaptive, etc.
  - Easily integrate sensor networks
    - Flexible message routing and filtering, sensor integration
  - Improve automation
    - Network "events" can trigger procedures
    - Automated reporting: sensors or health/status of spacecraft
- Simplified applications
  - Simplify use of distributed storage & processing
    - Data processed locally and shared efficiently
    - Software upload/installation (e.g., mods to Galileo s/w)
  - Assist failure discovery/recovery
    - Process restart or migration; application reconfiguration
  - Assist future autonomy
    - More information sources accessible for decisions
      - e.g., terrain, weather, off-board sensors
    - Simplify infrastructure for collaboration (joint planning, etc.)
      - Distributed intelligence; agents

N Lamarra 16

Example Future "Exploration Web"



N Lamarra 17

Directions for future work

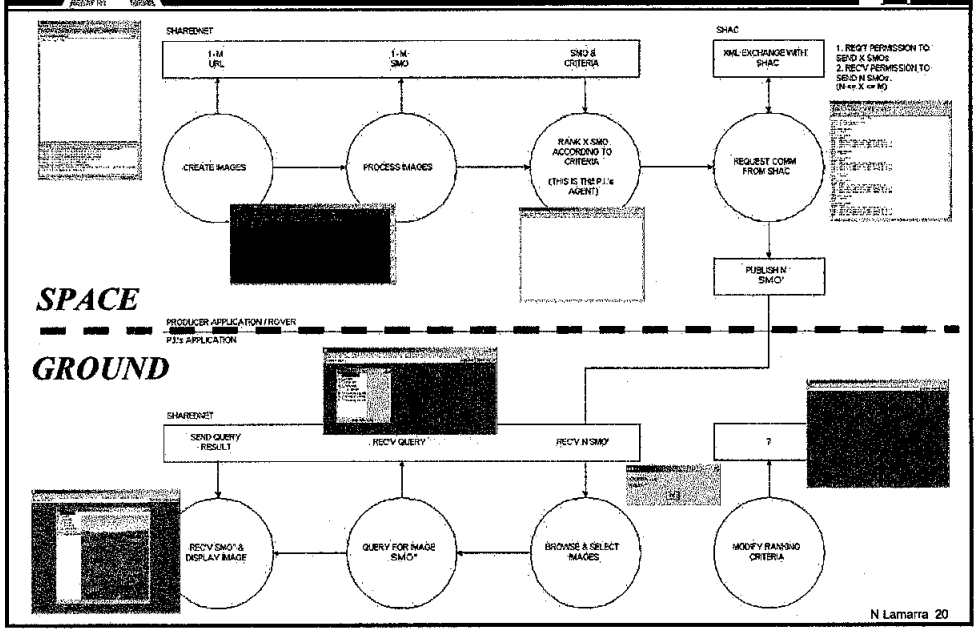
- Robust MOM is useful for almost all distributed apps
  - Show object information services layered above MOM
    - e.g., distributed data management: access/relay/archive/query
  - Show comm protocols layered below MOM
    - e.g., IP, CCSDS: PROX-1, CFDP
- Develop MOM functionality adapted for space
  - Simplify extending Message Object Model
  - Message forwarding; adaptive operation; scheduling
  - Verify robustness (disconnection, b/w, etc.)
  - Address CCSDS standards (e.g., SOIF)
  - Enable dynamic installation/removal – "standard services"
- Integrate with flight software
  - Simplify on-board processing (e.g., science extraction)
    - Provide support for dynamic algorithms (e.g., module upload)
  - Address software architecture issues (e.g., MDS, CLARAty)
- Migrate to RTOS on flight hardware
  - e.g., VxWorks on PPC

N Lamarra 18

Backup Charts

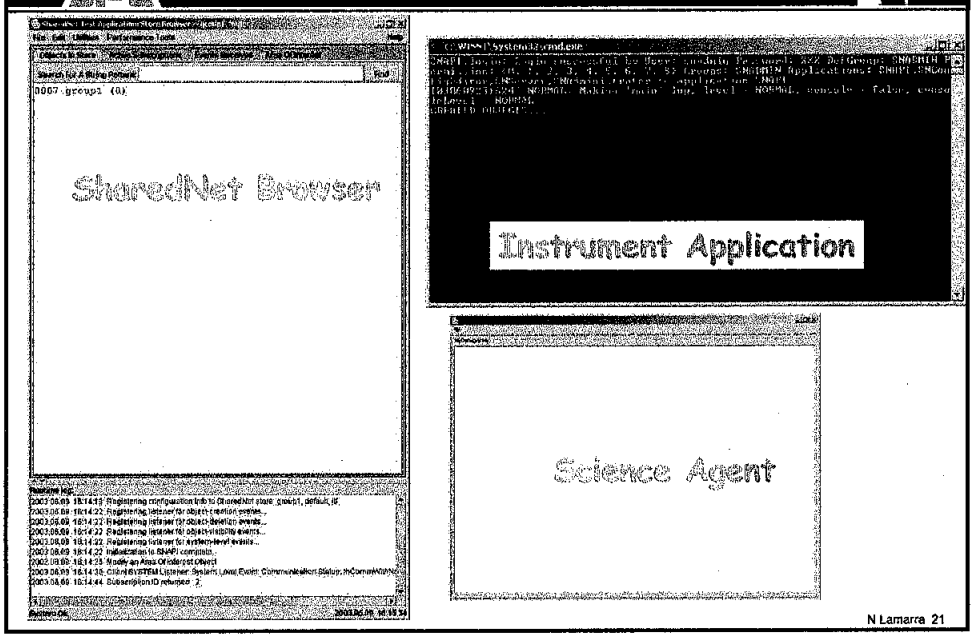
N Lamarra 19

Challenge Scenario Dataflow



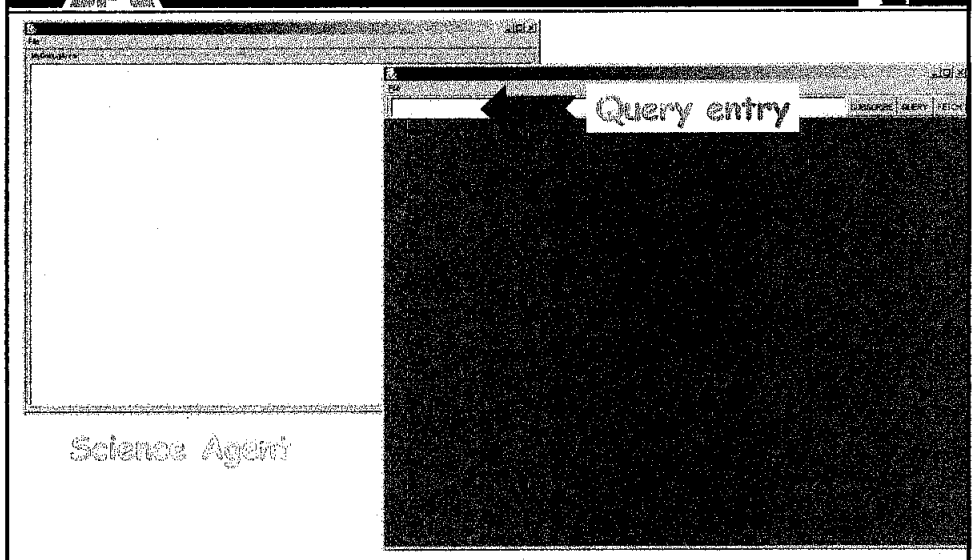
N Lamarra 20

Space: Remote Applications



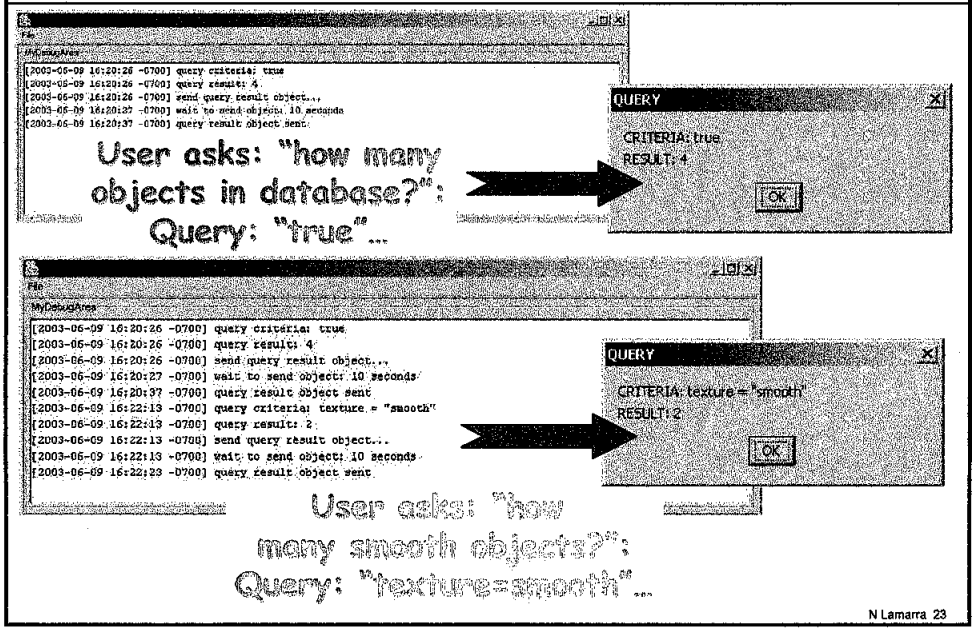
N Lamarra 21

Ground: User talks to Agent via UI



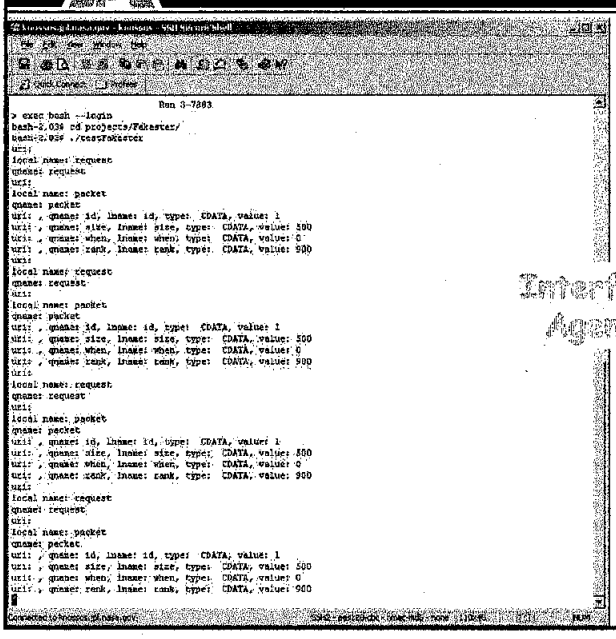
N Lamarra 22

Example Queries



N Lamarra 23

Science Agent requests downlink bandwidth



N Lamarra 24