FIRST | ESA | M | 0042.10

FIRST Integrated Network and Data Archive System

F I N D A S

End of Phase 1 Presentations
@ ESTEC -- 04/06/98

P. Estaria (SCI/PT)

AGENDA

09:00	Welcome	(P. Estaria/ESTEC)
09:10	Overview & Introduction	(P. Winder/VEGA)
09:25	User Requirements & Concept	(D. Crockford/VEGA)
10:20	*Coffee Break*	
10:35	Data Model and Architecture	(D. Crockford/VEGA)
11:30	Prototype Development	(P. Winder/VEGA)
12:10	Plans for Phase II	
12:30	*Lunch (1 hour)*	
13:30	Open Questions and Answers Session	
15:00	End	
15:00	Closed session (Kick-Off Phase II)	

FINDAS

- FINDAS is the backbone linking the 3 FIRST Instrument Control Centres (ICCs), the FIRST Science Centre (FSC) and the Mission Operation Centre (MOC) in the decentralised Ground Segment Architecture adopted for FIRST
- July '96 -- FINDAS concept defined
- October '96 - March '97 -- Various reviews; concept endorsed; prototype recommended
- August '97 -- Open Invitation to Tender (ITT) issued to Industry
- January '98 Contract (300 KECU; FFP) awarded to VEGA - K.O.

FINDAS PROTOTYPE

- Overall duration : 1 year
  - Phase 1 (Analysis Phase) : 4 months
  - Phase 2 (Implementation Phase) : 8 months
- Prototype delivered to ESTEC : January-February '99
- Consolidation Phase
  - Warranty period (3 months TBD)
  - User trials
  - Draft ICD (June '99)

Introduction

FINDAS Phase I Review Presentation

ESA/ESTEC · 4 June 1998

Phil WINDER, VEGA

Project team

Personnel – VEGA Group PLC

- **Phil Winder** – project manager
- **Derek Crockford** – technical engineer
- Derek Greer – domain expert (ISO)
- development and test engineers

Timescale – FINDAS <u>Prototype</u> Project

- analysis – 4 months
- implementation and specification – 8 months

Motivation

FINDAS – *backbone* of decentralised ground segment architecture

Key FINDAS Concepts

- seamless transition
- data distribution
- security and configuration control
- use of COTS tools
- large data volume
- extended lifetime – scalability, flexibility, adaptability
- object-oriented approach

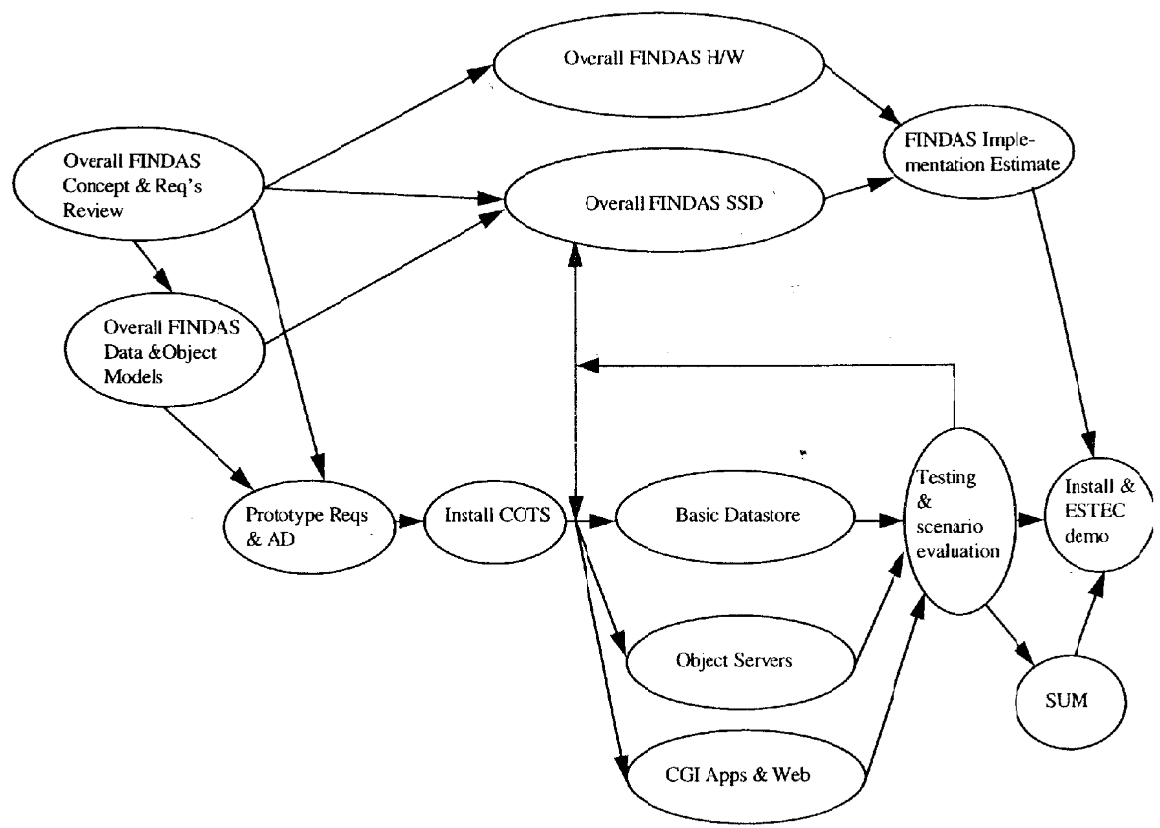
Prototype Goals

- assess feasibility of concept
- add missing requirements
- identify risk areas and cost drivers
- explore critical issues
- implementation & demonstration
- COTS tools & methods
- specification of full system
- foundation for later phases
- **demonstrate added value**

Project Parameters

- FINDAS is *infrastructure* for distributed ground segment – input from ICCs/FSC/MOC/DPCs required to develop client applications & middleware, and to refine data schema
- Generic support to PLANCK from prototype – no specific requirements nor analysis of PLANCK data model

Work Breakdown



Progress So Far

Achievements:

- critical review of requirements and proposed concept
- definition of major parts of overall FINDAS data schema
- analysis of architecture and design options
- proposals for implementation of key elements including development environment and COTS tools

Status:

- end of Phase I

Presentation Schedule

- Introduction – Phil Winder
- Analysis 1: High level – Derek Crockford
- Analysis 2: Technical aspects – Derek Crockford
- Prototype Development – Phil Winder
- Plans for Phase II – Phil Winder
- Open Discussion

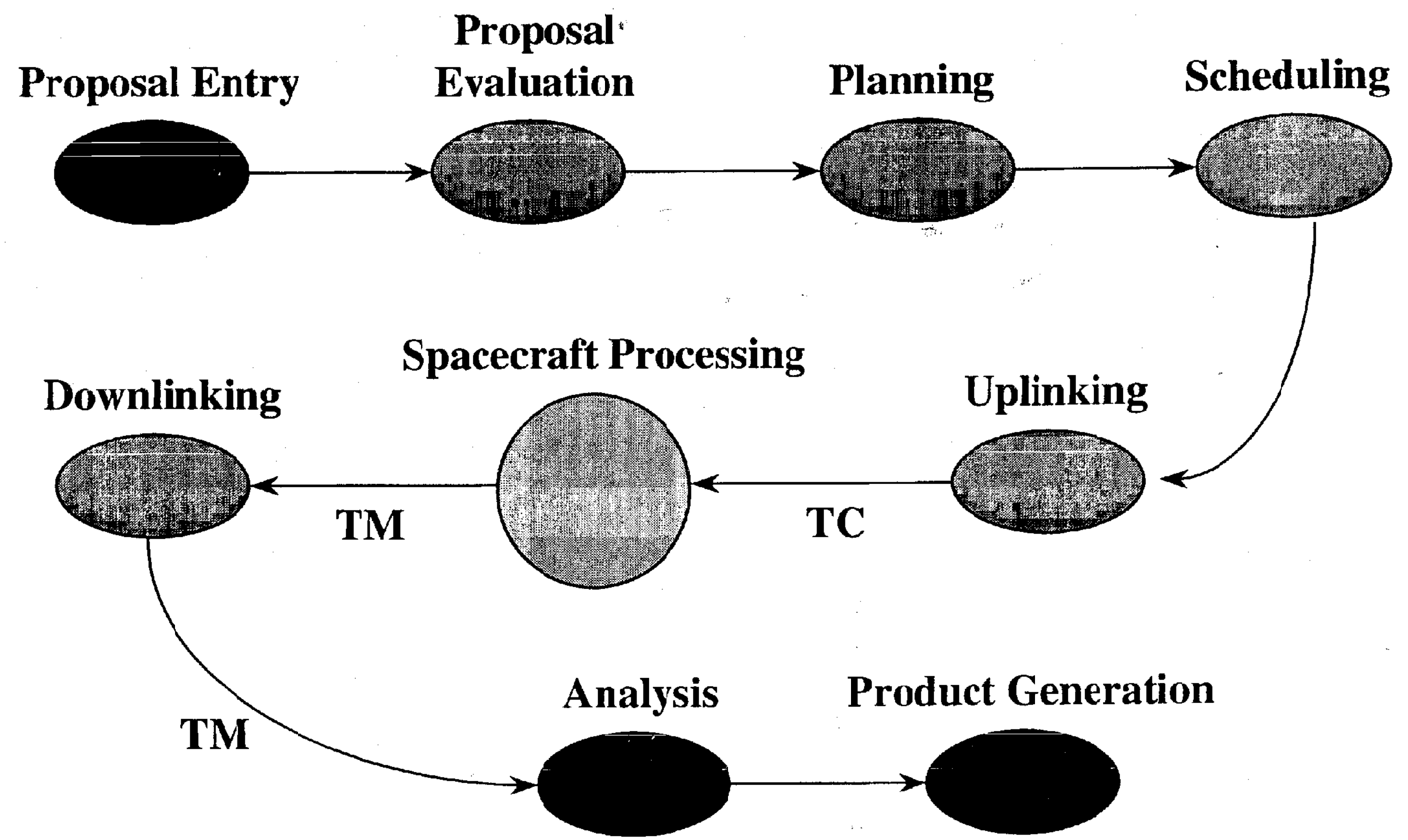
FINDAS Prototype: Analysis (1)

Phase I Review Presentation

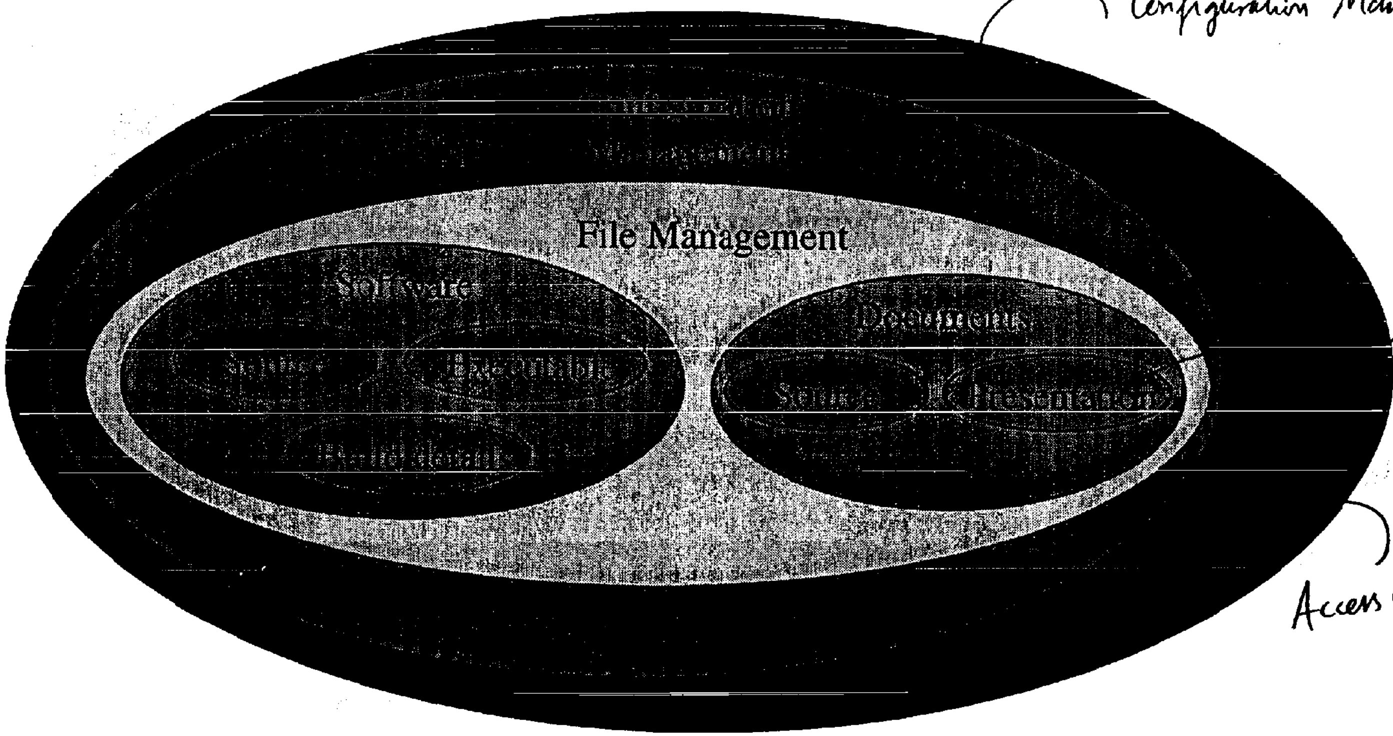
ESA/ESTEC

4 June 1998

What is a Ground Segment? - Processing Requirements



Other Ground Segment Requirements



Configuration Management

eg Pdf's

Access Control

Why not just keep ISO?

- Software was developed separately and then had to be integrated into a centralised operations centre – leading to changes and inconsistencies

- The heterogeneous software systems had their own separate databases, which could not be kept consistent

- Changes to one subsystem or data structure often required matching changes to other subsystems hence causing a ripple effect

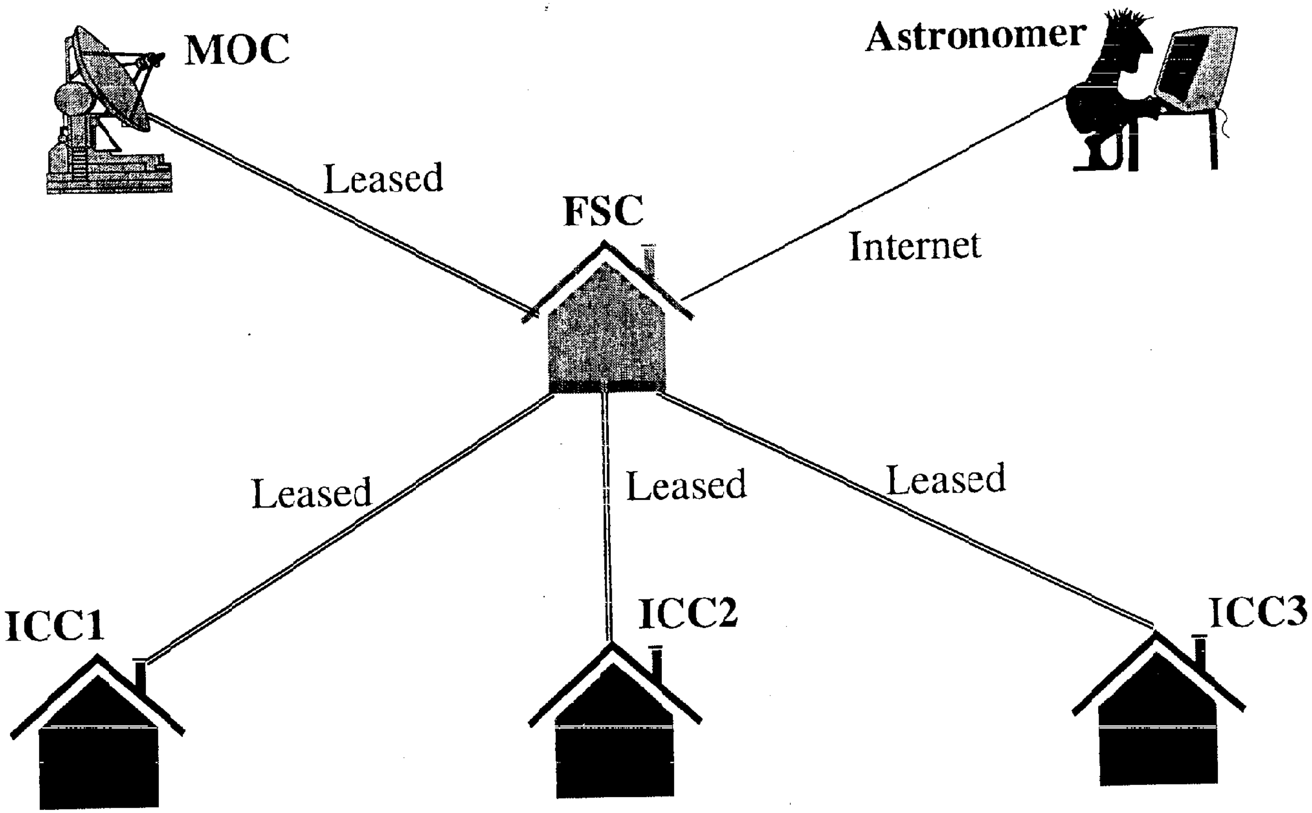
- Considerable effort was required from the instrument groups throughout all mission phases – the centralised operations required collocated instrument expertise during the operational mission phase, which could have been more effectively utilised at the home institute

- Interfaces had to be changed (sometimes extensively) between different mission phases – leading to respecification and reimplementation of parts of the ground segment

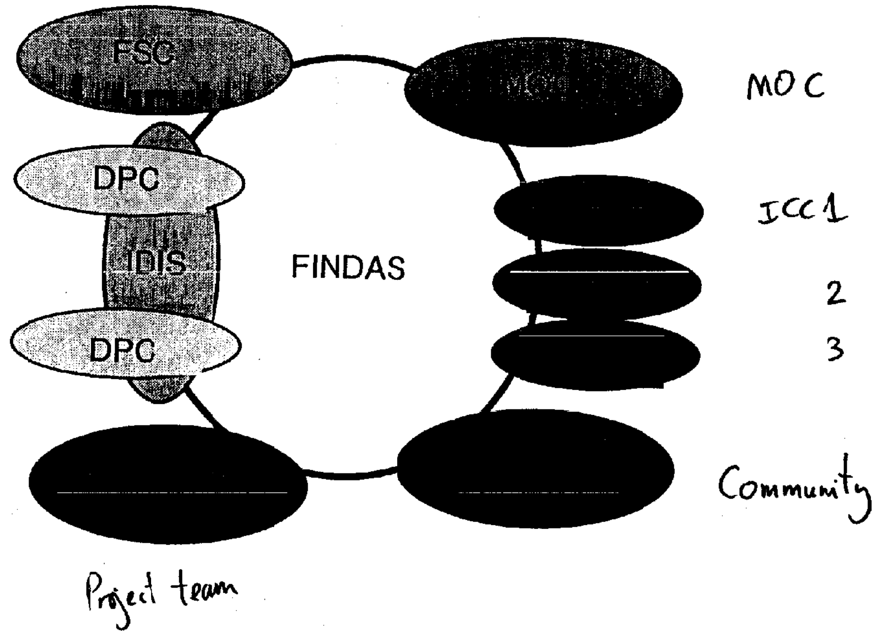
A Contrast between ISO and FIRST

ISO	FIRST
Continuous TC/TM	No contact for 22hrs/day
All TM real time	Mostly dump TM
Instrument babysitting required	Autonomy & macro-commands
Fixed TM format (TDF)	Packet based TM
TM processing via pipeline	No pipeline (TBC)
Centralised ground segment	Distribution to ICCs
Data on different databases	Common database
Disjoint mission phases	Smooth transition essential
Little configuration control	Configuration control required
No document management	Document management included
Separate entry of observations, calibrations & tests	Unification using proposals

FIRST Ground Segment
- Physical Layout



FIRST Ground Segment - FINDAS Concept



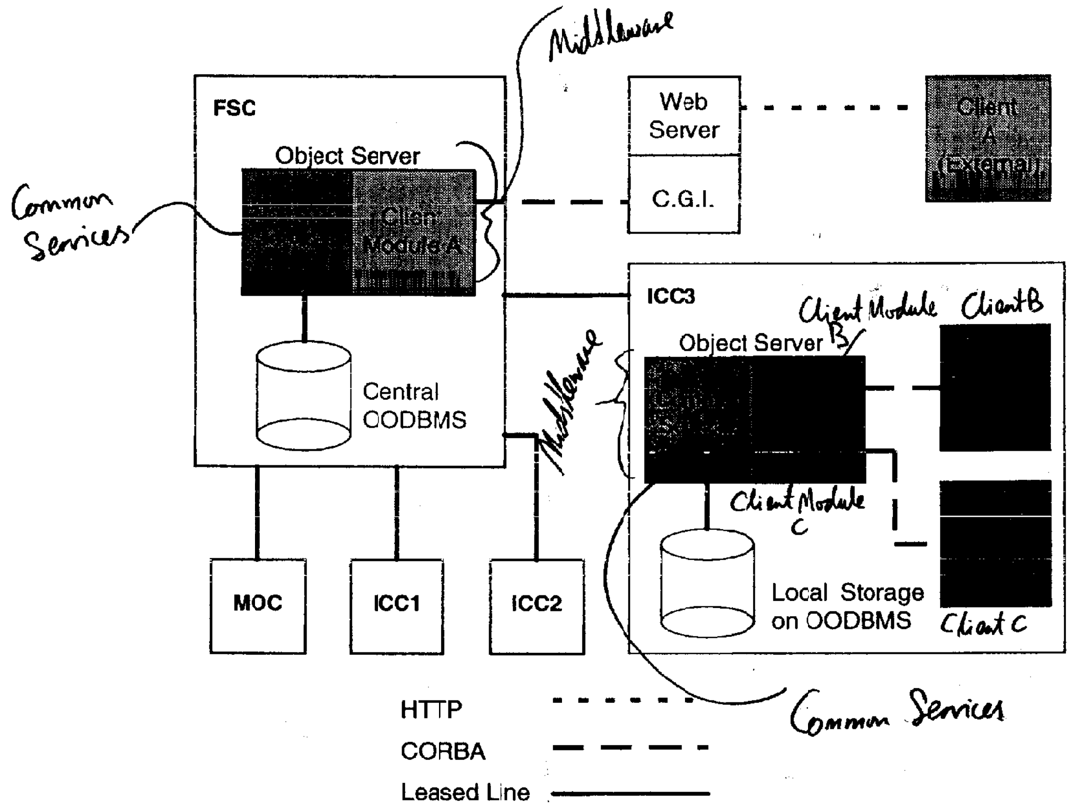
Critical Issues for FINDAS

- Distribution of ground segment processing across many sites
- Seamless transitions between mission phases - this is essential in a project with limited resources and tight schedules
- Scalability - final archive size will be of the order of 2.5 Tbytes
- Adaptability, flexibility - needs to meet evolving requirements over 20 year lifetime
- Durability - no part of system must depend on a single vendor and migration paths must exist to likely new technologies
- Robustness - system must be resistant to failures and misconfiguration
- Real time performance - real time TM must reach analysis stations quickly enough for a response to be made within the contact window
- Ability to handle sophisticated data schemas
- Unified configuration management

FINDAS Design Decisions

- The adoption of object oriented technology wherever appropriate
- A solution based upon a COTS object-oriented database management system compliant with the ODMG standard
- User applications which connect to the database by means of a 3-tier client server architecture
- A distributed architecture in which the master database is present at the FSC with replicated/selected data storage at each ICC
- The use of the CORBA distributed object standard for communication between middleware and clients
- Clients based upon web browser / HTML technology (use of Java applets included implicitly)

Summary of FINDAS Architecture



FINDAS Benefits Matrix (1)

Issue	Implementation	Benefit
Single database	Master database at FSC	Coordination of interface definitions, configuration control & access rights
Distributed architecture	Replicated / selective storage at each ICC & good communications	ICC work performed at home institute in all phases
Applications developed locally in all phases	3-tier client server architecture	Minimise integration issues
Flexibility / adaptibility	3-tier client server architecture	Provide customised views and localise the effects of change
Seamless transitions between mission phases	Common data interfaces and object oriented techniques	Minimise effort on unnecessary interface changes

FINDAS Benefits Matrix (2)

Issue	Implementation	Benefit
Development cost	Use of COTS whenever practicable	Minimise development effort
Complex data items with sophisticated relationships	Object oriented database management system (standard feature)	Implementation directly reflects desired organisation
Configuration management	Object oriented database management system (standard feature)	Reliable configuration management
20 year lifetime	Adherence to standards	Assure migration path
External access / post mission archive	External web access to scalable database is part of architecture	No separate development effort required
Ability to query database (associative access)	OQL querying for expert users; basic querying for novices	Flexible querying according to level / privilege of user

FINDAS Benefits Matrix (3)

Issue	Implementation	Benefit
Document management	Part of database system	Consistent access to documents by all authorised users
Software management	Part of database system	Consistent access to software by all authorised users
No pipeline processing	Astronomers download package of TM, auxiliary data & processing software	No expensive processing burden on FIRST
Unified system for entry of observations, calibrations & tests	All entry via single proposal generation system	Simple, consistent interface for all users

FINDAS Prototype: Analysis (2)

Phase I Review Presentation

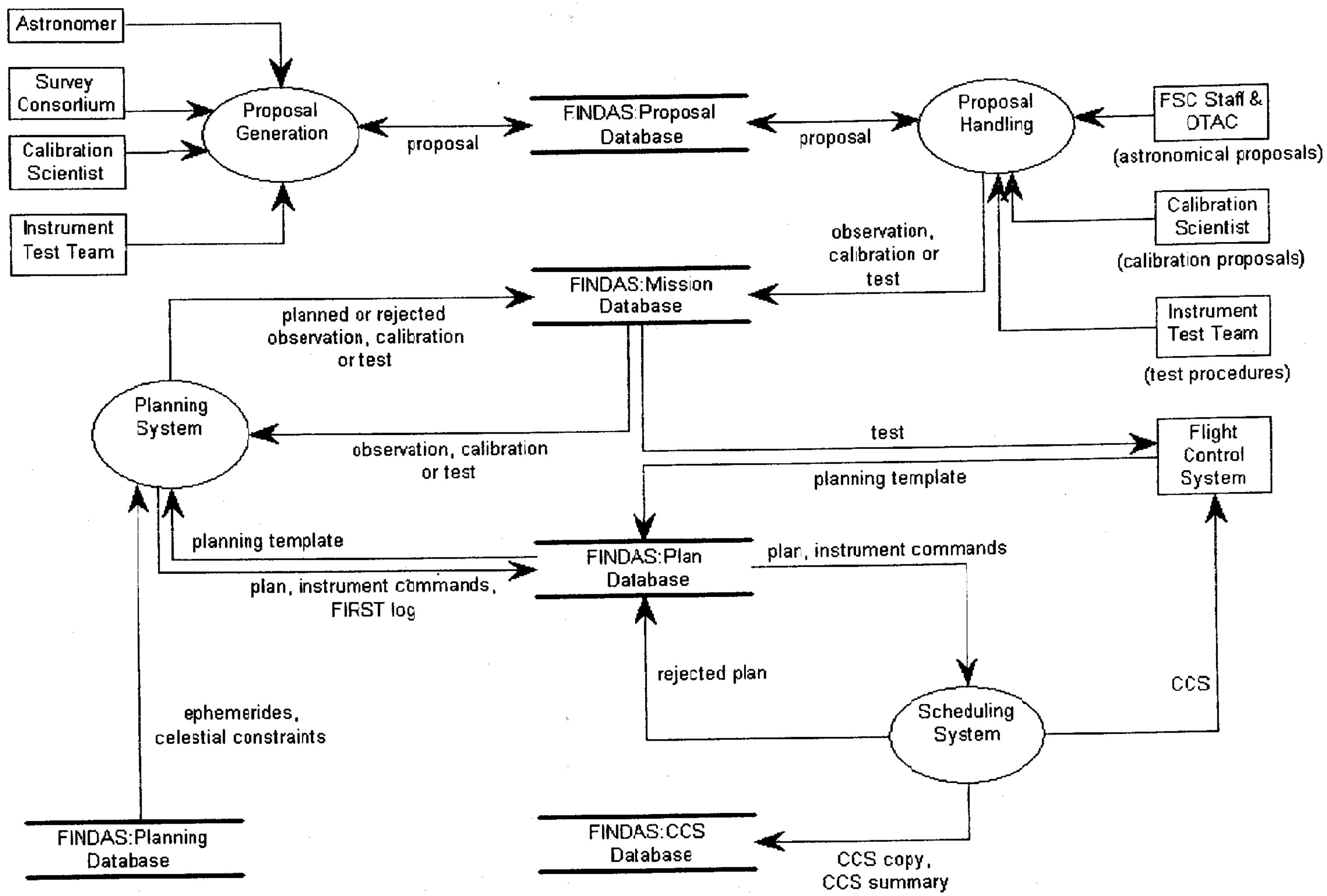
ESA/ESTEC

4 June 1998

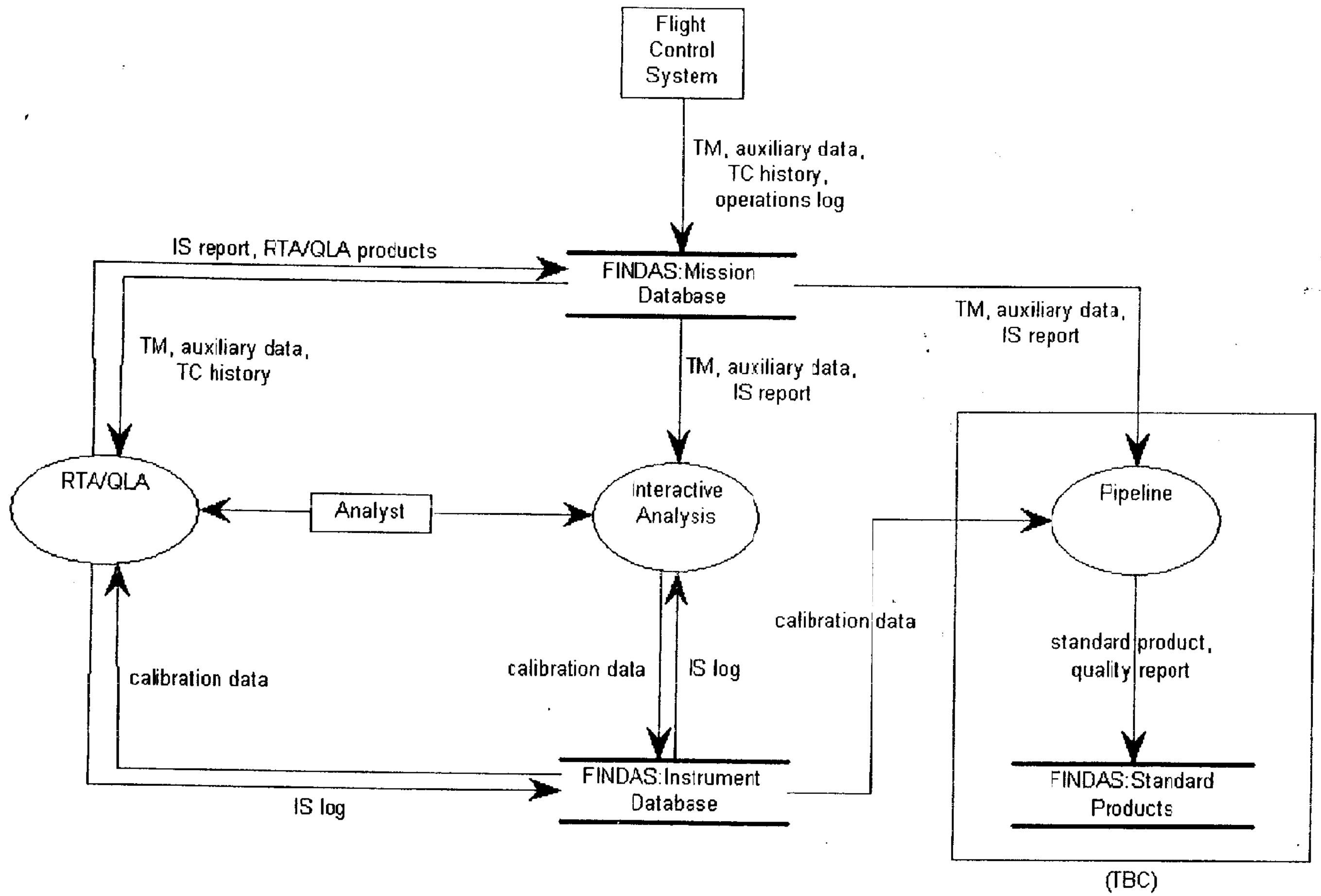
Overview

- Functionality of FIRST ground segment: dataflow diagrams
  - uplink and downlink
  - operational and ILT phases
- FINDAS database schema: object oriented techniques
  - some example classes
  - uplink and downlink
  - operational and ILT phases
  - instrument & spacecraft views
- System design
  - COTS tool framework
  - 3-tier client server architecture
  - Local storage options
- Configuration management
  - example
  - the Version class

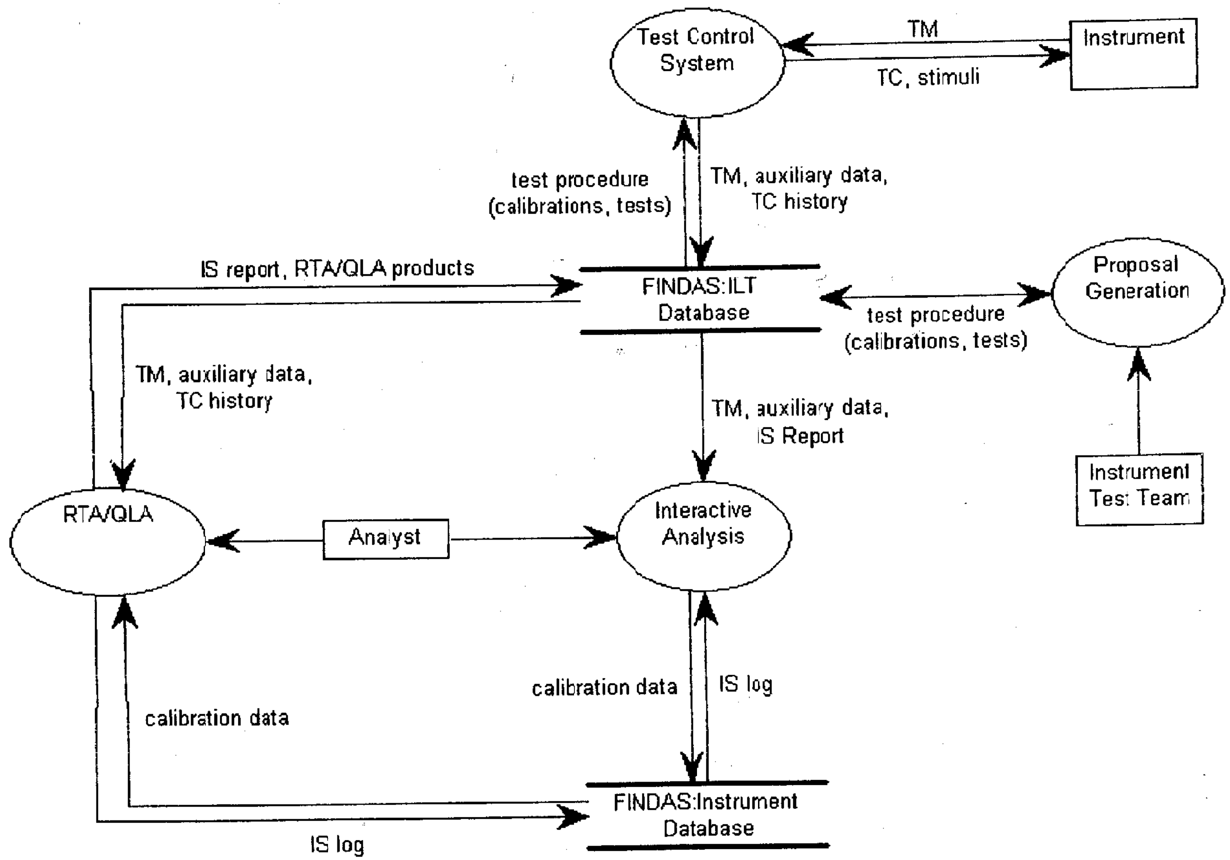
Uplink Dataflow: Operational Phase



Downlink Dataflow: Operational Phase



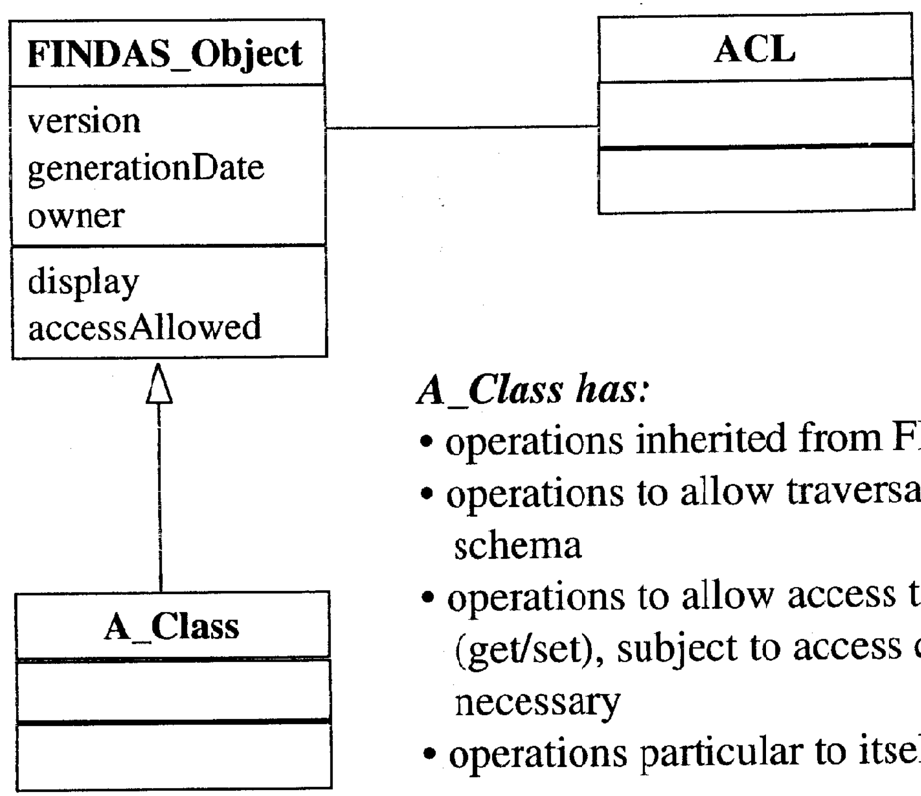
Dataflow for the ILT Phase



Object Oriented Development

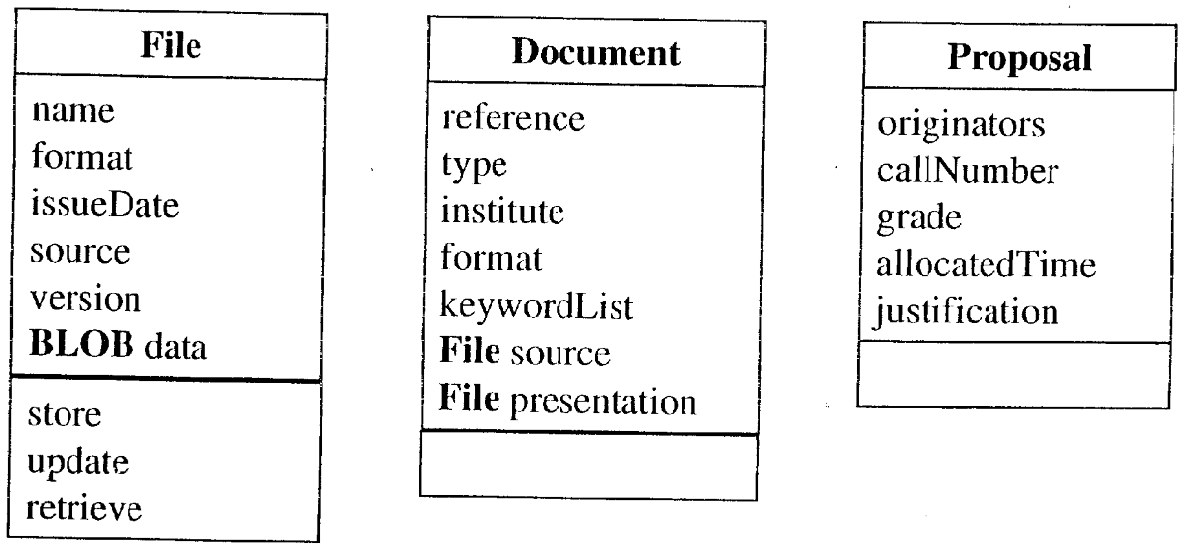
- handling of complexity: both data and functionality are organised into a single structure
- information hiding: provision of interfaces at an appropriate level of abstraction
- allows specialization through inheritance and polymorphism, while maintaining a common interface unchanged
- provides natural level of granularity for access control and configuration management
- classes can be defined with the aim of providing a particular view of a set of other classes
- abstract base classes and handle classes (or "Cheshire Cat" classes) can be used to decouple interfaces from implementations
- extensive COTS tool support available

The FINDAS_Object Class

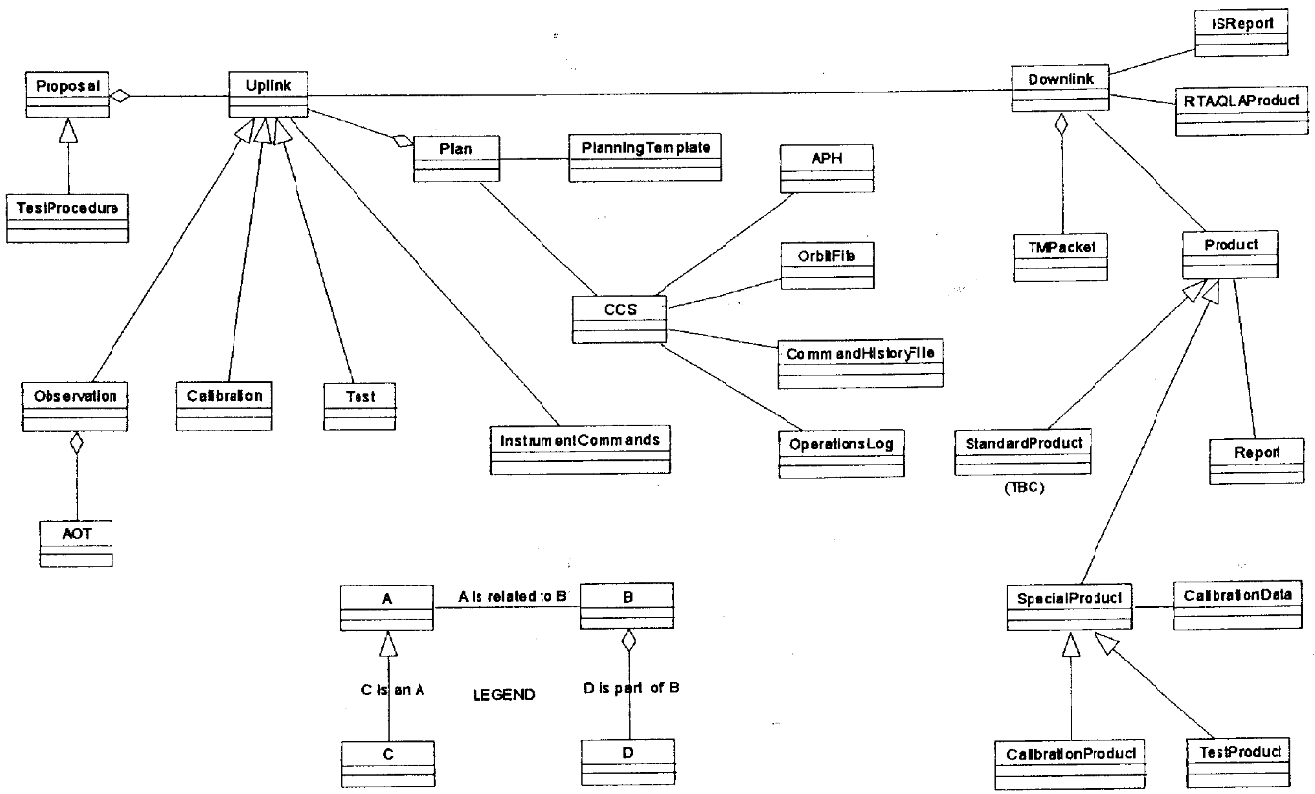


A_Class has:
- operations inherited from FINDAS_Object
- operations to allow traversal of database schema
- operations to allow access to attributes (get/set), subject to access control where necessary
- operations particular to itself

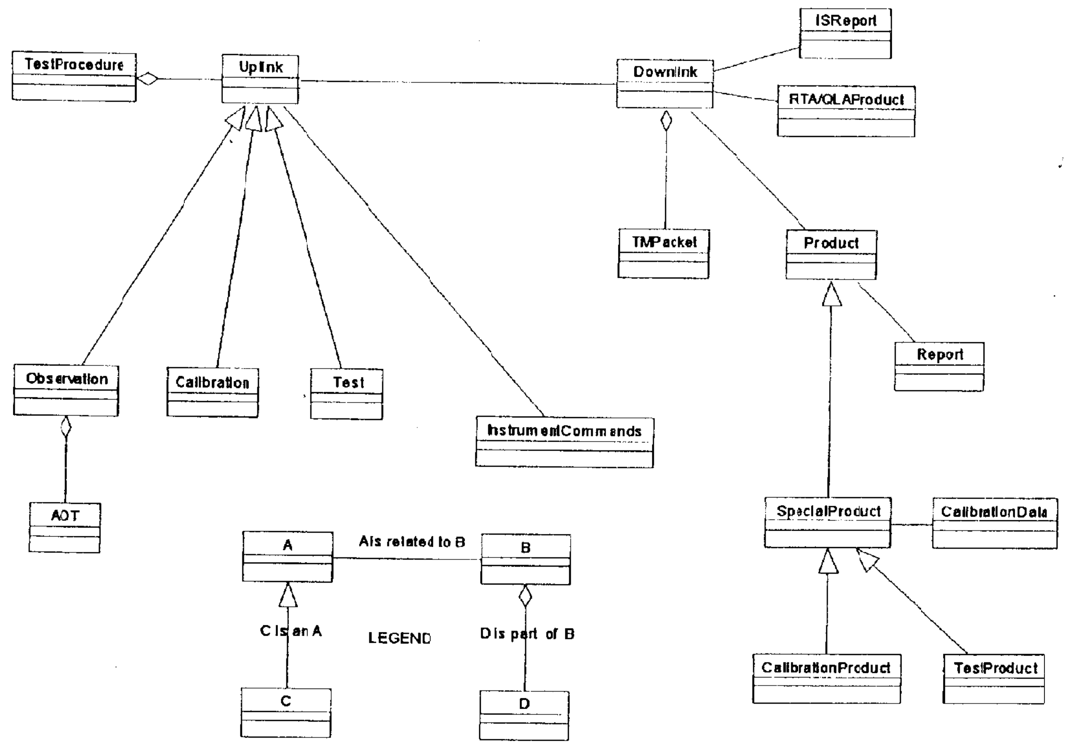
Example FINDAS Classes



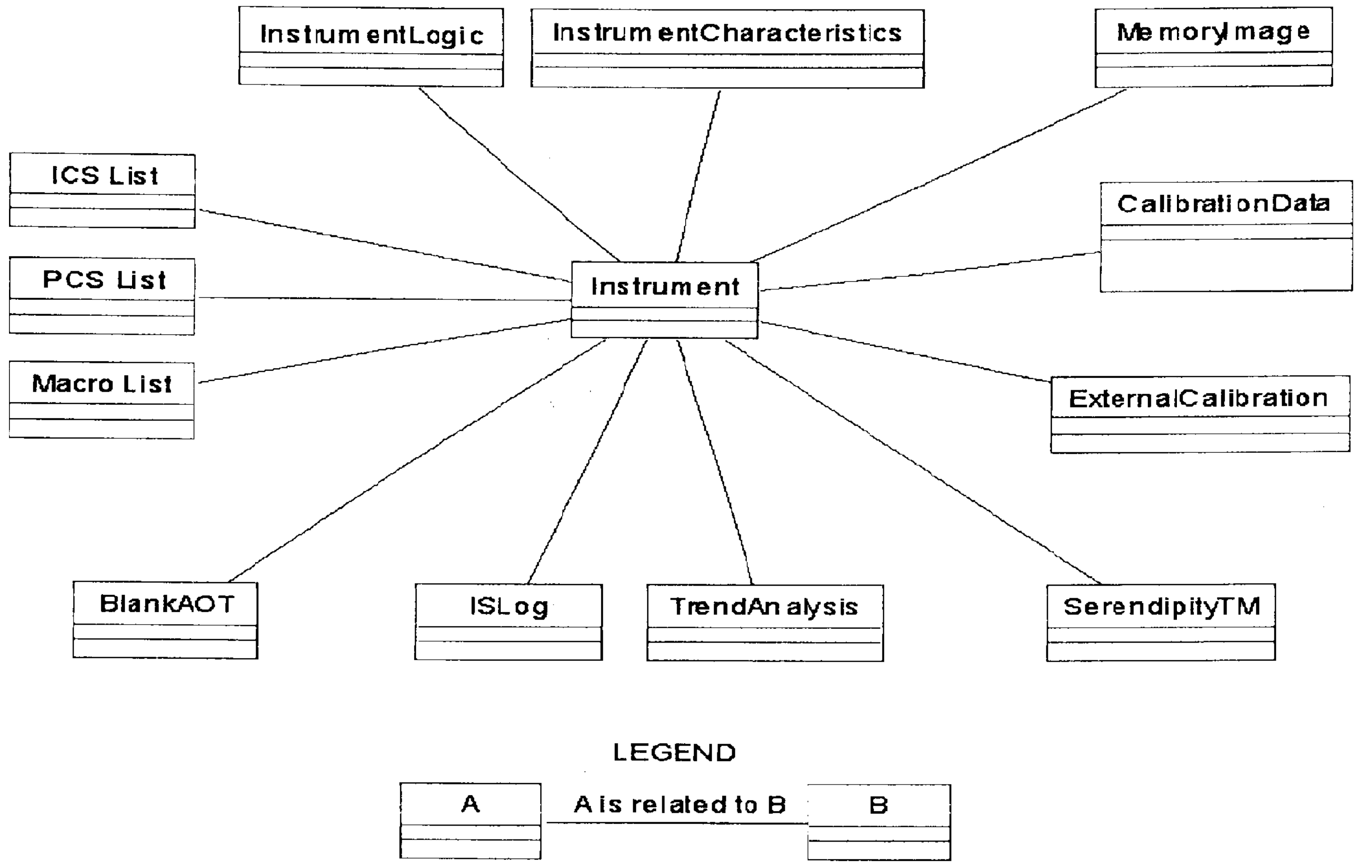
Class Diagram for the Operational Phase



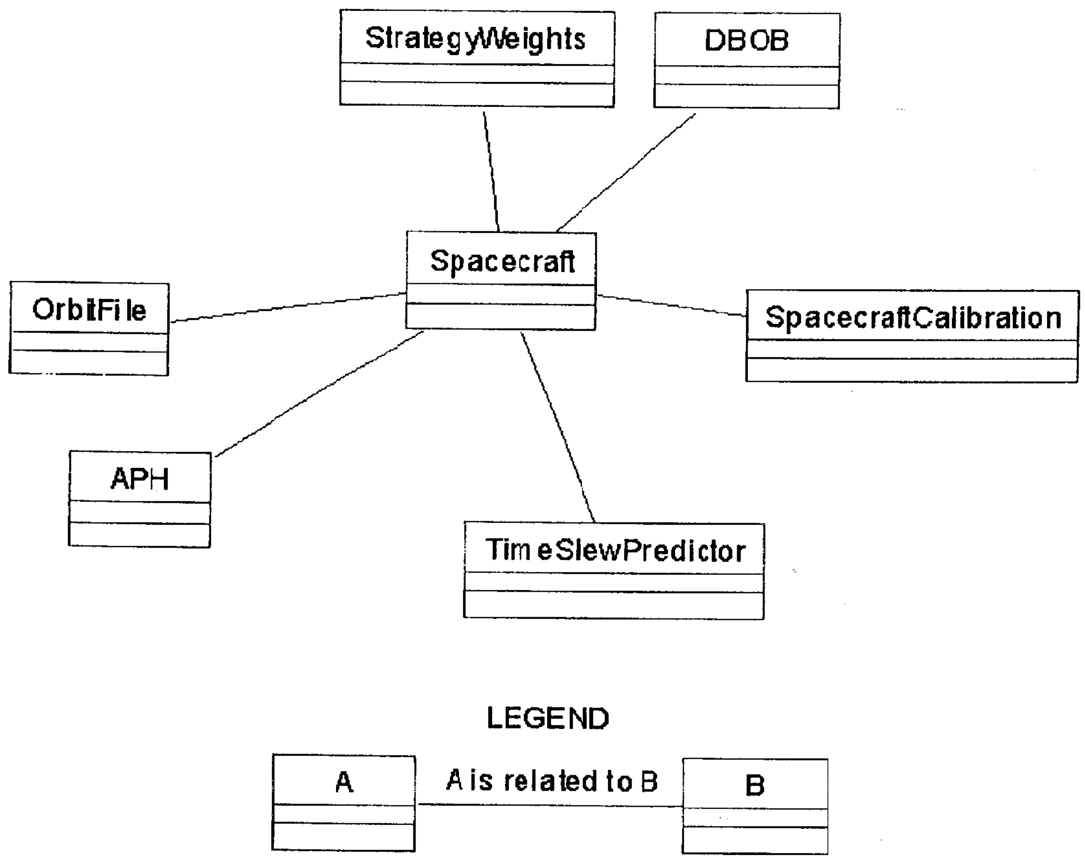
Class Diagram for the ILT Phase



Class Diagram Centred on Instrument



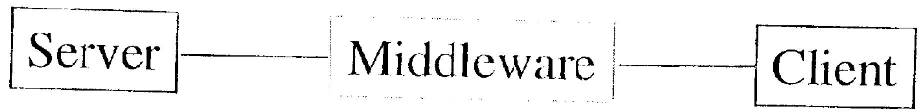
Class Diagram Centred on Spacecraft



COTS Framework

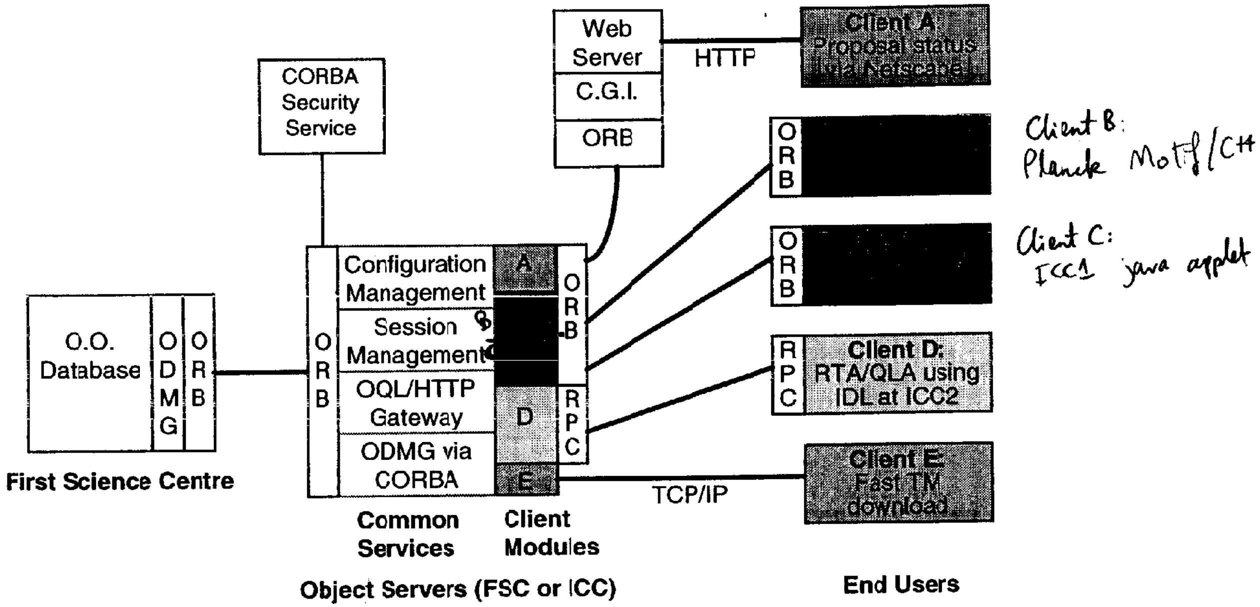
- COTS tools => reduced development effort and industrial strength solutions
- Choose object oriented database
  - provides high performance traversal of object relationships
  - provides direct mapping of object model
- ODMG compliant database
  - standards-based, can substitute database, follow migration paths
  - near transparent database access from object oriented languages using ODMG bindings
- Database access via OQL => standards-based, powerful querying using syntax familiar to SQL users, conforming to type system of chosen object oriented language
- Network connectivity using CORBA
  - standards-based
  - language and platform independence via IDL (Interface Definition Language)
  - additional COTS functionality available in the form of CORBA Services (and later, CORBA Facilities)
  - migration to use of DII/DSI over longer term (?) (Dynamic Interfaces)

3-Tier Client Server Architecture

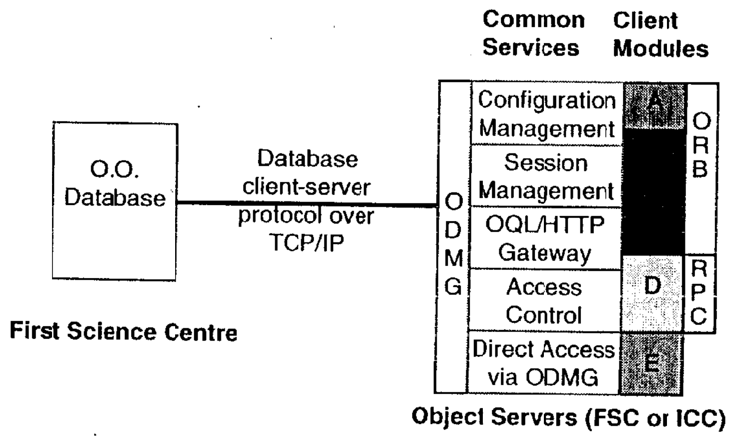


- Structure/vendor of database is abstracted behind middleware
- Middleware provides logical view of data which is tailored to each particular application
- Basic interfaces available to all, but specialised interfaces can be provided at middleware by inheritance
- Performance can be enhanced by having multiple copies of a given middleware process
- Fine grained access control can be enforced by middleware
- Facilitates specialised configuration management
- Flexible: new object servers can be added easily and old ones removed from the system
- Much fewer middleware machines than client machines => major modifications affect relatively small number of machines
- Middleware can provide transparent management of local storage

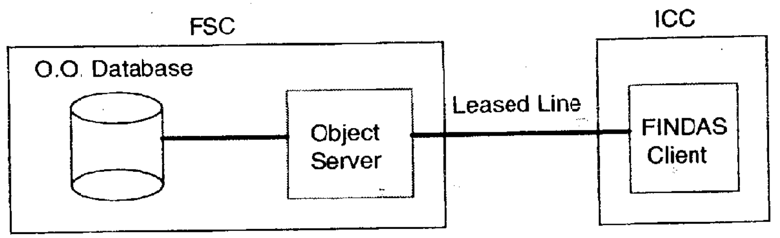
Client Server Option 1: CORBA Back-End



Client Server Option 2: ODMG Back-End



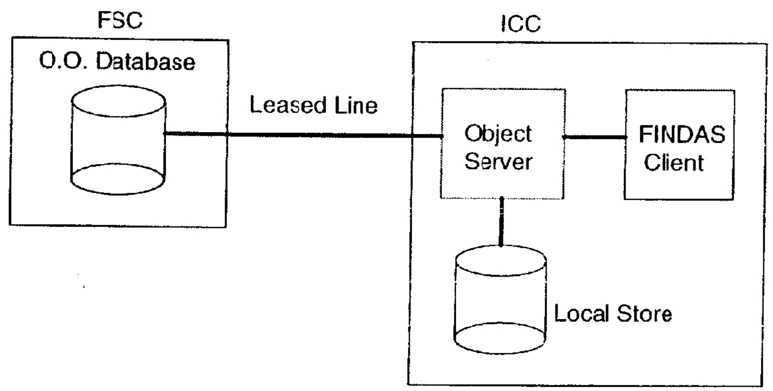
Distribution Option 1: No Local Storage at ICC



+ minimum infrastructure/support at ICC

- heavy loading of leased line
- reliability of leased line critical

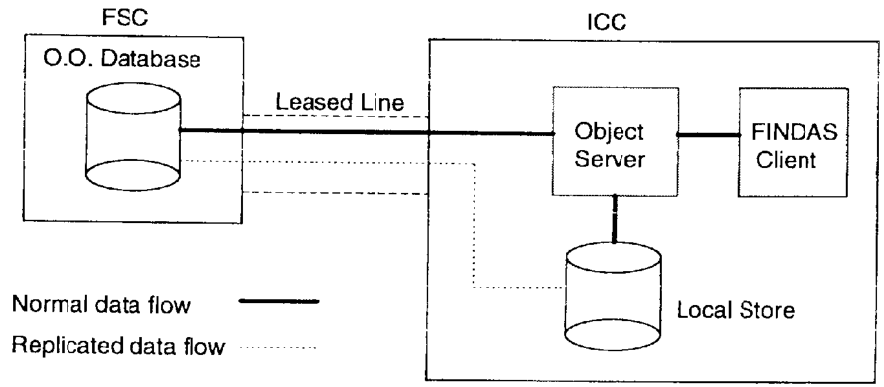
Distribution Option 2: Selective Storage at ICC



+ some independence from leased line (TBC)
+ more efficient bandwidth utilization

- some commitment required from ICC

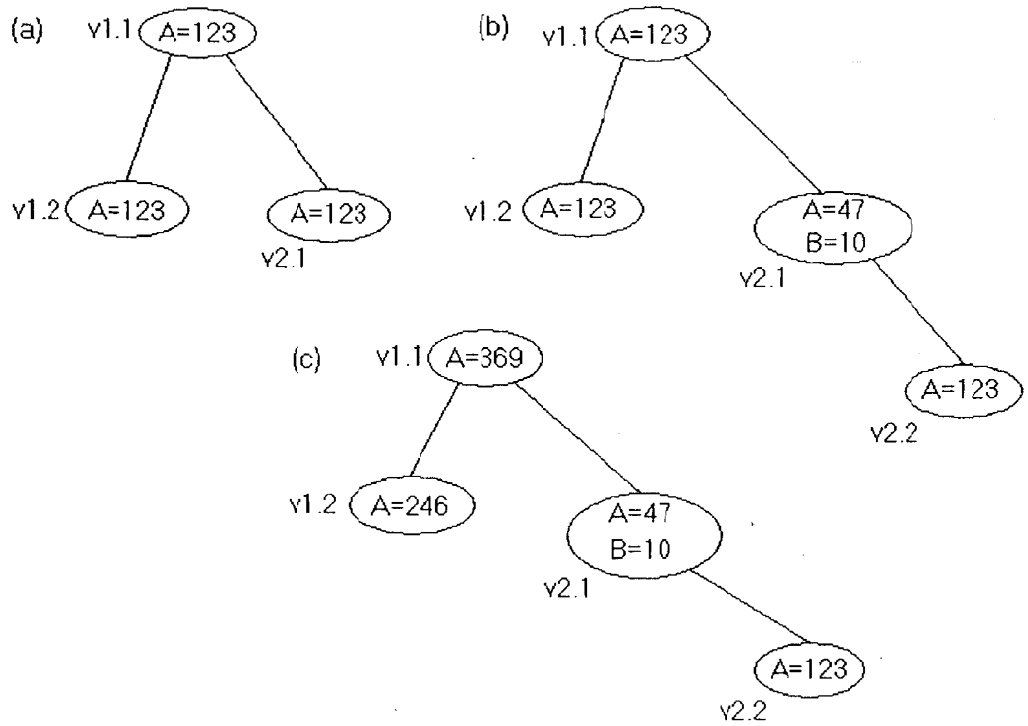
Distribution Option 3: Replicated Storage at ICC



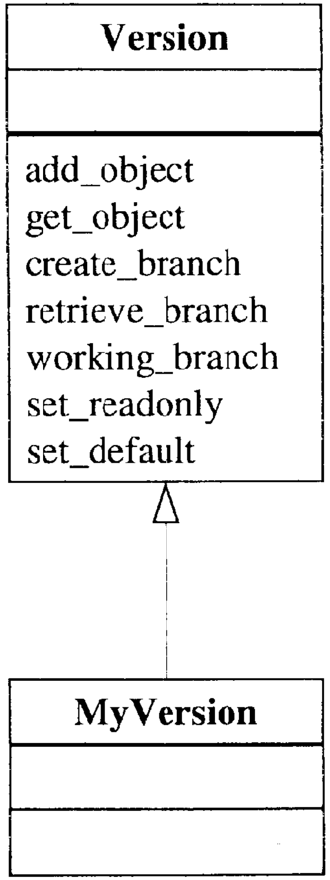
+ some independence from leased line (TBC)
+ more efficient bandwidth utilization
+ less development effort

- some commitment required from ICC
- restriction on deleting from local store

Configuration Management Example



The Version Class



- Basic class, available to all applications
  - add/access object contained in branch
  - create/retrieve branches
  - set working branch
  - set a branch to be read-only or default

- Specialised class for particular application
  - record creator/creation time of branch
  - special authorisation required to define default branch
  - default branch is read-only

FINDAS Prototype Development

FINDAS Phase I Review Presentation

ESA/ESTEC · 4 June 1998

Phil WINDER, VEGA

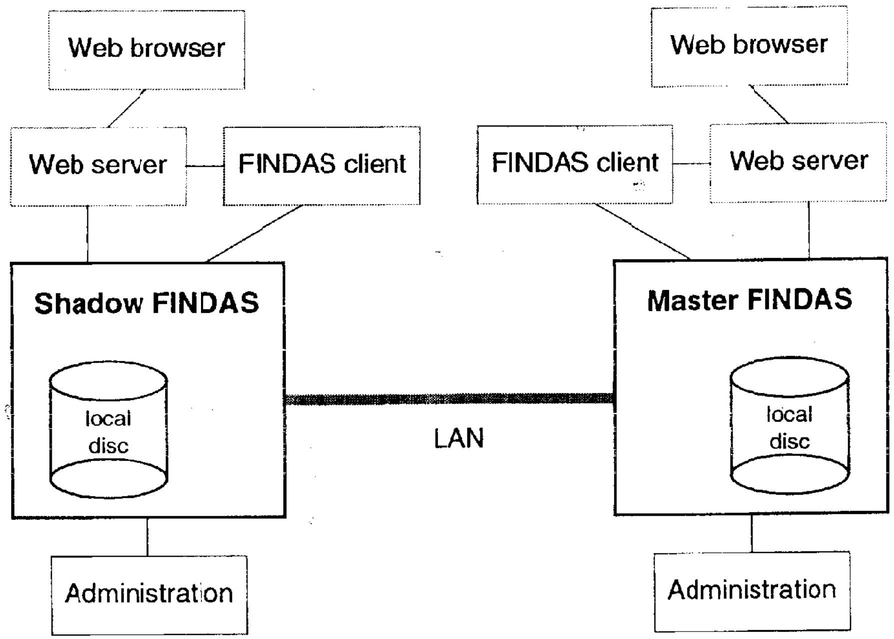
Purpose of Prototype

- demonstrate feasibility and attractiveness
- encourage participation
- investigate risk areas & cost drivers
- scope full FINDAS development
- foundation for baseline operational system

Requirement/Function

Requirement		Function	
Seamless transition	(✓)	User categories	✗
Distributed architecture	✓	Configuration control	✓
Data size and scalability	✗	Web user interface	✓
Schema complexity	(✓)	Link to DMS	✓
Adaptability, flexibility, durability	✓	Document management system	✓
Data access & security	(✓)	Database administration	✓
Real-time performance	✓	Published client API	(✓)
System availability	✗	Event notification	(✓)
CORBA client API	✓	FTP cache	✗
CORBA vs. ODMG	(✓)	Selective data off-lining	✗
Robustness to network failure	(✓)		

Demonstration Infrastructure



Demonstration Scenarios

- schema evolution
- multiple views
- real-time transfer
- access DMS via URL
- internal document management system
- event notification

- remote access and shadow database
- robustness to network error
- logging of transactions
- CORBA vs. ODMG
- configuration control
- third-party client (Not yet agreed: requires client API)

Schema evolution/multiple views

- modify database schema – preserve client interface
- eager vs. lazy database update
  - → update each instance when it is referenced
  - ↳ update all instances in one go
- alternative interfaces to single object

∴ *painless evolution and seamless transition*

Real-time telemetry transfer

- real-time transfer between source and sink
- measure overheads and network efficiency

- demonstrate robustness to network error

- support off-line bulk update if required

∴ *efficient and reliable data distribution*

Remote access and Shadow d'base

- transparent and secure manipulation of remote master database
- efficient client access via local shadow database

∴ *distribution strategy improves response and network efficiency but is hidden from client application*

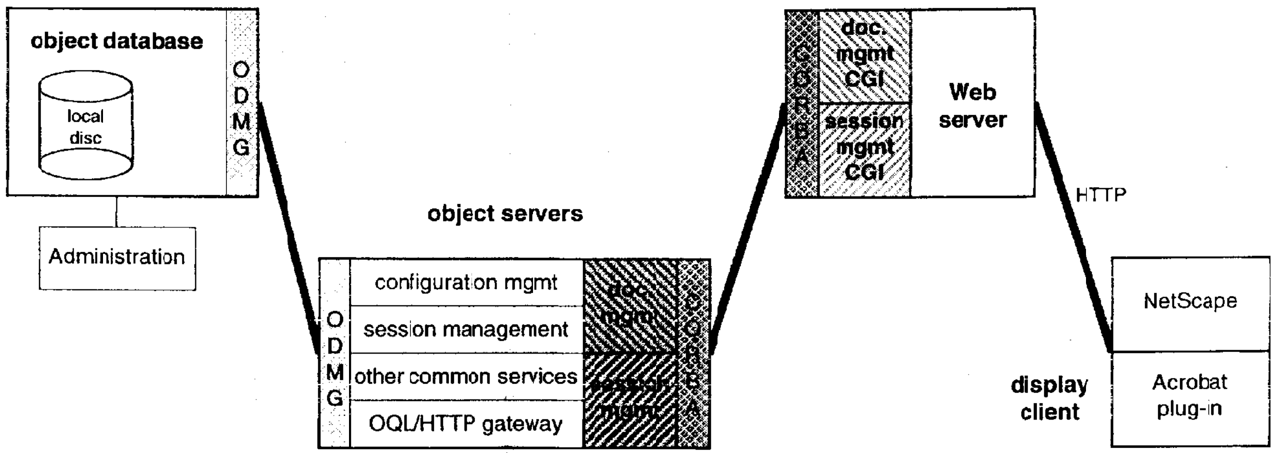
Document management

- data format: PDF plus source
- Web-based navigation and keyword search
- document display
- document entry and revision

∴ *exercise FINDAS infrastructure*

∴ *bare-bones document management capability*

Architecture



FINDAS document management system

Choice of Database System

- diverse datasets and complex relationships
- on-going modification of links between items
- long-term evolution of data schema
- efficiency and scalability

⇒object-oriented database (OODBMS)

n.b. hidden from client application.

O_2 object-oriented database

- scalable and efficient client/server architecture
- existing large databases
- user base
- support for replication
- built-in Web interface
- standards compliance
- flexibility and evolution
- interoperability – e.g. CORBA, ORACLE …
- database utilities
- supplementary modules available

Development environment

- UML
- SPARC Solaris workstations
- C++
- O_2
- CORBA
- CGI & Web browser

Conclusions

we have

- identified topics
- outlined scenarios
- illustrated design
- specified tools

we will

- reduce risk
- establish validity of concept
- feed into full FINDAS
- deliver to ESTEC

Plans for Phase II

FINDAS Phase I Review Presentation

ESA/ESTEC · 4 June 1998

Phil WINDER, VEGA

Tasks for Phase II

Phase I – *completed*

- critical review of requirements and proposed FINDAS concept
- definition of major parts of overall FINDAS data schema
- analysis of architecture and design options
- proposals for prototype development and COTS

During Phase II

- implement prototype (including document management system)
- document prototype (SUM, STD, SVVP)
- install at ESTEC (including COTS) & demonstrate scenarios
- overall FINDAS hardware+software specification

Following Phase II

At end of Phase II (February 1999)

- foundation for implementation of operational FINDAS

→ prototype source code and COTS at ESTEC – "hands-on" experimentation

→ review organisational approach to distributed system

→ analyse PLANCK requirements

→ consider domain-specific aspects and application development