



University
of Glasgow

Dynamic and Static Testing

Prof. Chris Johnson,
School of Computing Science, University of Glasgow.
johnson@dcs.gla.ac.uk
<http://www.dcs.gla.ac.uk/~johnson>

- The processes used during
 - validation and verification.
- White and black boxes.
- Static and Dynamic techniques
- Case Study...

- Black box tests:
 - tester has no access to information
 - about the system implementation.
- Good for independence of tester.
- But not good for formative tests.
- Hard to test individual modules...

- White box tests:
 - tester can access information about
 - the system implementation.
- Simplifies diagnosis of results.
- Can compromise independence?
- How much do they need to know?

- Module testing:
 - tests well-defined subset.
- Systems integration:
 - tests collections of modules.
- Acceptance testing:
 - system meets requirements?
- Results must be documented.

Changes will be costly.

- Functional testing:
 - test cases examine functionality;
 - see comments on verification.
- Structural testing:
 - knowledge of design guides tests;
 - interaction between modules...
 - test every branch (coverage)?
- Random testing:
 - choose from possible input space;
 - or beyond the "possible"...

- Dynamic testing:
 - execution of system components;
 - is environment being controlled?
- Static testing:
 - investigation without operation;
 - pencil and paper reviews etc.
- Most approaches use both.
- Guide the test selection by using:
 - functional requirements:
 - safety requirements (see previous lecture).

Lifecycle phase	Dynamic testing	Static testing
Requirements analysis and specification		x
Top-level design		x
Detailed design		x
Implementation	x	x
Acceptance testing	x	

- Where do you begin?
- Look at the original hazard analysis;
 - demonstrate hazard elimination?
 - demonstrate hazard reduction?
 - demonstrate hazard control?
- Must focus both on:
 - expected and rare conditions.
- PRA can help - but for software?

- All of this will cost time and money.
 1. Review test plans.
 2. Recommend tests based on the hazard analyses, safety standards and checklists, previous accident and incidents, operator task analyses etc
 3. Specify the conditions under which the test will be conducted.
 4. Review the test results for any safety-related problems that were missed in the analysis or in any other testing.
 5. Ensure that the testing feedback is integrated into the safety reviews and analyses that will be used in design modifications.
- Must be planned, must be budgeted.

- Partitioning:
 - identify groups of input values;
 - do they map to similar outputs?
- Boundary analysis:
 - extremes of valid/invalid input.
- Probabilistic Testing:
 - examine reliability of system.
- (State) Transition tests:
 - trace states, transitions and events.

- Simulation:
 - assess impact on EUC (IEC61508).
- Error seeding:
 - put error into implementation;
 - see is test discover it (dangerous).
- Performance monitoring:
 - check real-time, memory limits.
- Stress tests:
 - abnormally high workloads?

- Boundary conditions.
- Incorrect and unexpected inputs sequences.
- Altered timings - delays and over-loading.
- Environmental stress - faults and failures.
- Critical functions and variables.
- Firewalls, safety kernels & other safety features.
- Usual suspects...automated tests?

- Cannot test all software paths.
- Cannot even test all hardware faults.
- Not easy to test in final environment.
- User interfaces very problematic:
 - effects of fatigue/longitudinal use?
 - see section on human factors.
- Systems **CHANGE** the environment!
- How can we test for rare events? 10^9 years?



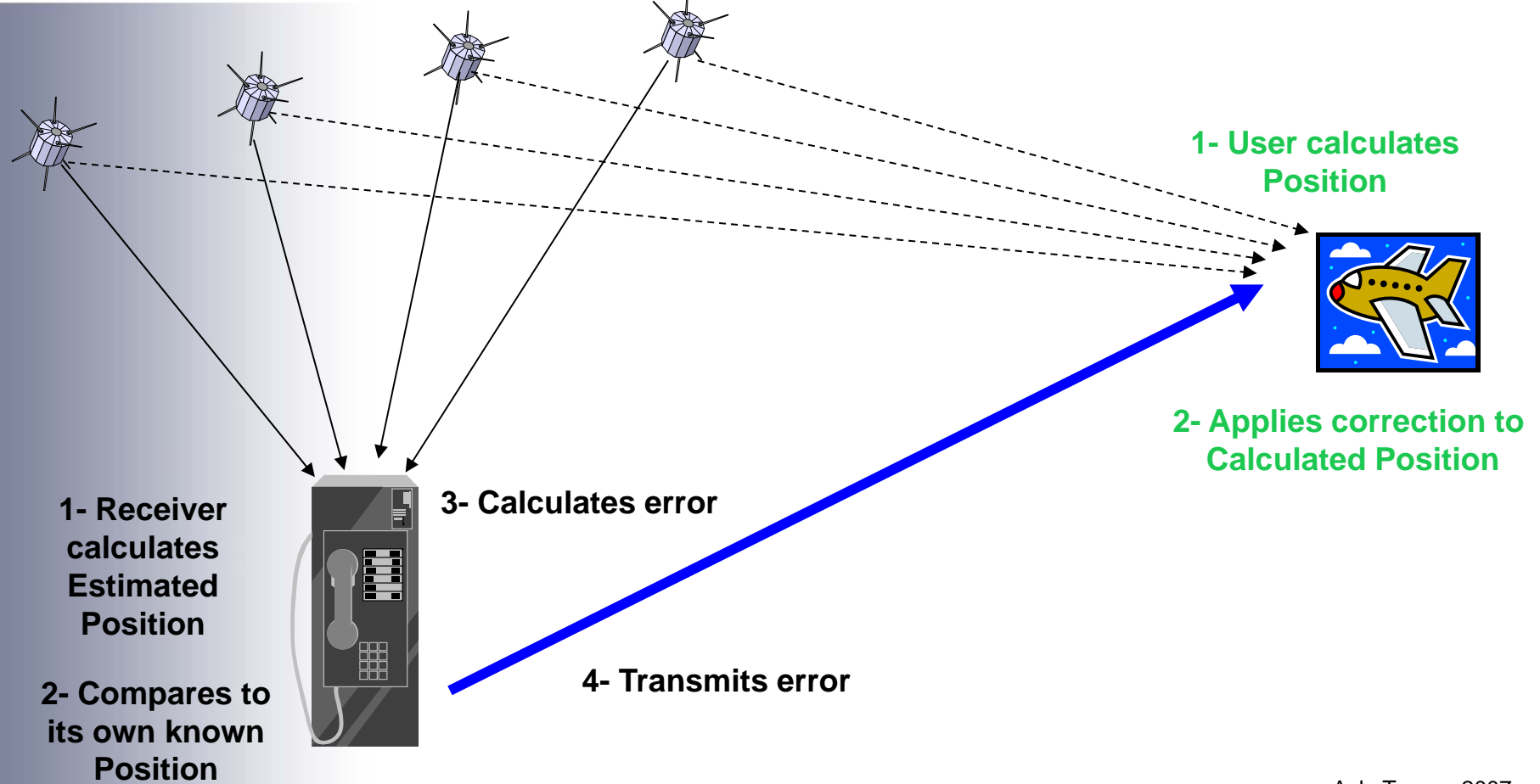
Testing can prove the presence of errors, but not their absence.

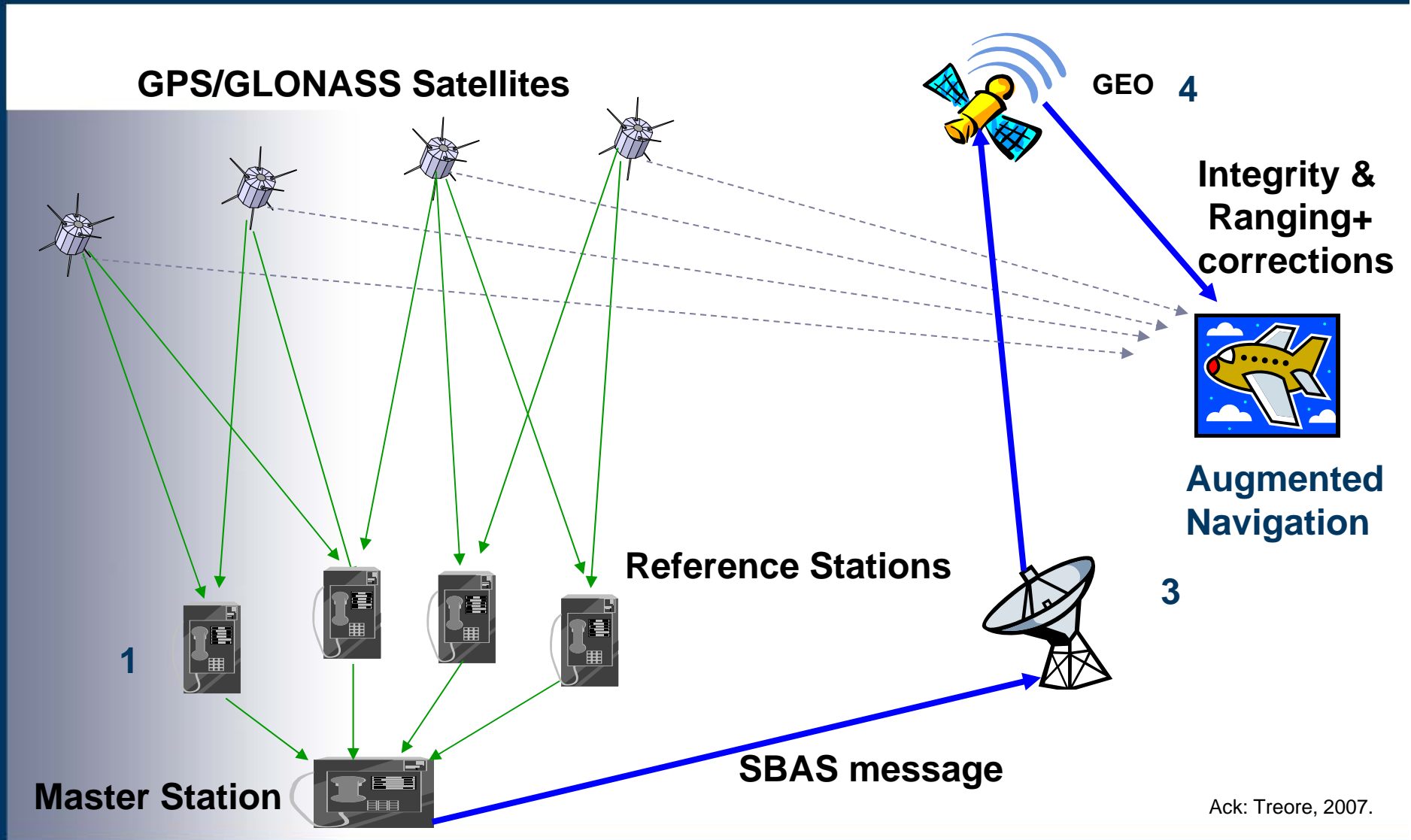
- Don't test the system itself.
- Test an abstraction of the system
- Perform checks on requirements?
- Perform checks on static code.
- Scope depends on representation...

- Walkthroughs:
 - peer review by other engineers.
- Fagan inspections:
 - review of design documents.
- Symbolic execution:
 - use term-rewriting on code;
 - does code match specification?
- Metrics:
 - lots (eg cyclomatic complexity);
 - most very debatable...

- Sneak Circuit Analysis:
 - find weak patterns in topologies;
 - for hardware not software.
- Software animation:
 - trace behaviour of software model;
 - Petri Net animation tools.
- Performance/scheduling theory:
 - even if CPU scheduling is static;
 - model other resource allocations.
-
- Formal methods cf 00-60 with DO-178B...

GPS/GLONASS Satellites With Pseudolite Augmentation





- **Accuracy.**
 - How correct is the aircraft position estimate.
- **Integrity.**
 - Largest aircraft position error without detection;
- **Availability.**
 - How often systems give desired Accuracy/
Integrity;
- **Continuity.**
 - Probability that operation can be completed.



Safety of Life requirements

■ ICAO SARPS high-level integrity requirements on Signal In Space

Typical Operation	Horizontal Alert Limit	Vertical Alert Limit	Integrity	Time to alert	Continuity	Availability
En-route	2 NM	N/A	1×10^{-7} /h	15 s	1×10^{-4} /h to 1×10^{-8} /h	0.999 to 0.99999
En-route (terminal)	1 NM	N/A	1×10^{-7} /h	15 s		
Ininitial approach, NPA departure	0.3 NM	N/A	1×10^{-7} /h	10 s		
APV-I	40.0 m	50 m	$1-2 \times 10^{-7}$ /app (150s)	10 s	1×10^{-6} /h in any 15s	
APV-II	40.0 m	20 m	$1-2 \times 10^{-7}$ /app (150s)	6 s		
CAT I	40.0 m	15-10 m	$1-2 \times 10^{-7}$ /app (150s)	6 s		

Three EGNOS Services



- **Open Service**

- Free service started October 2009.

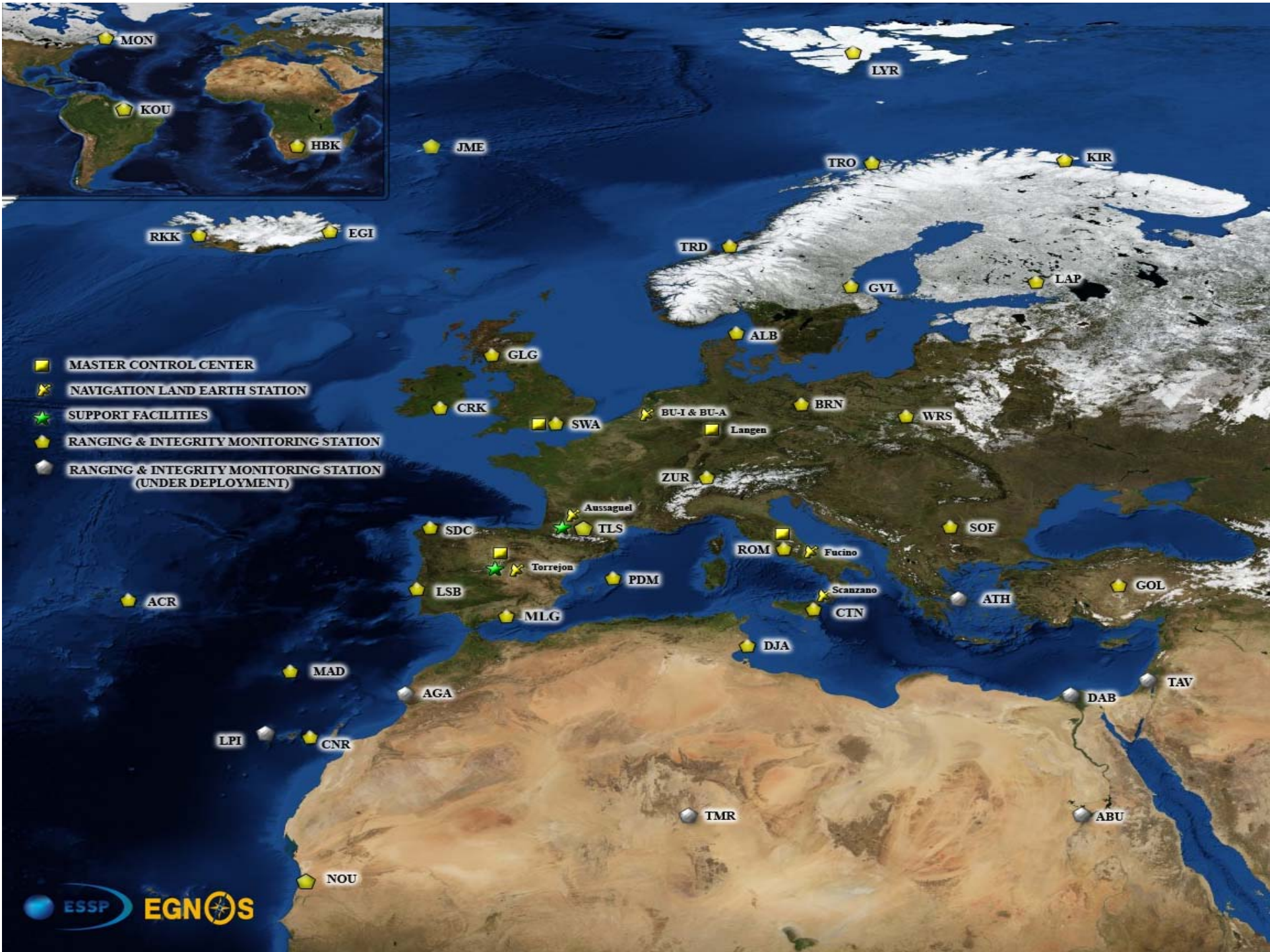
- **Safety-of-Life Service (SoL).**

- For safety-critical industries certified against Single European Sky/ICAO requirements 2011.

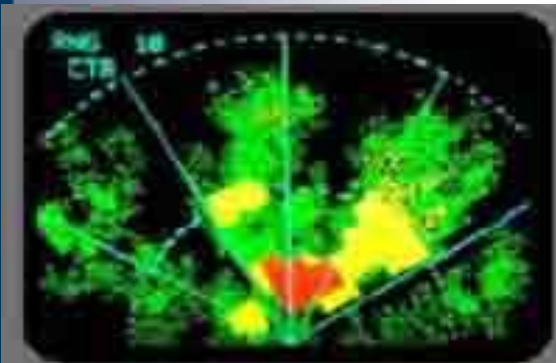
- **EGNOS Data Access Server (EDAS):**

- Terrestrial commercial data disseminates through non-GEO means, EGNOS data within performance boundaries in real time supporting professional market.





- 40% of losses Controlled Flight into Terrain

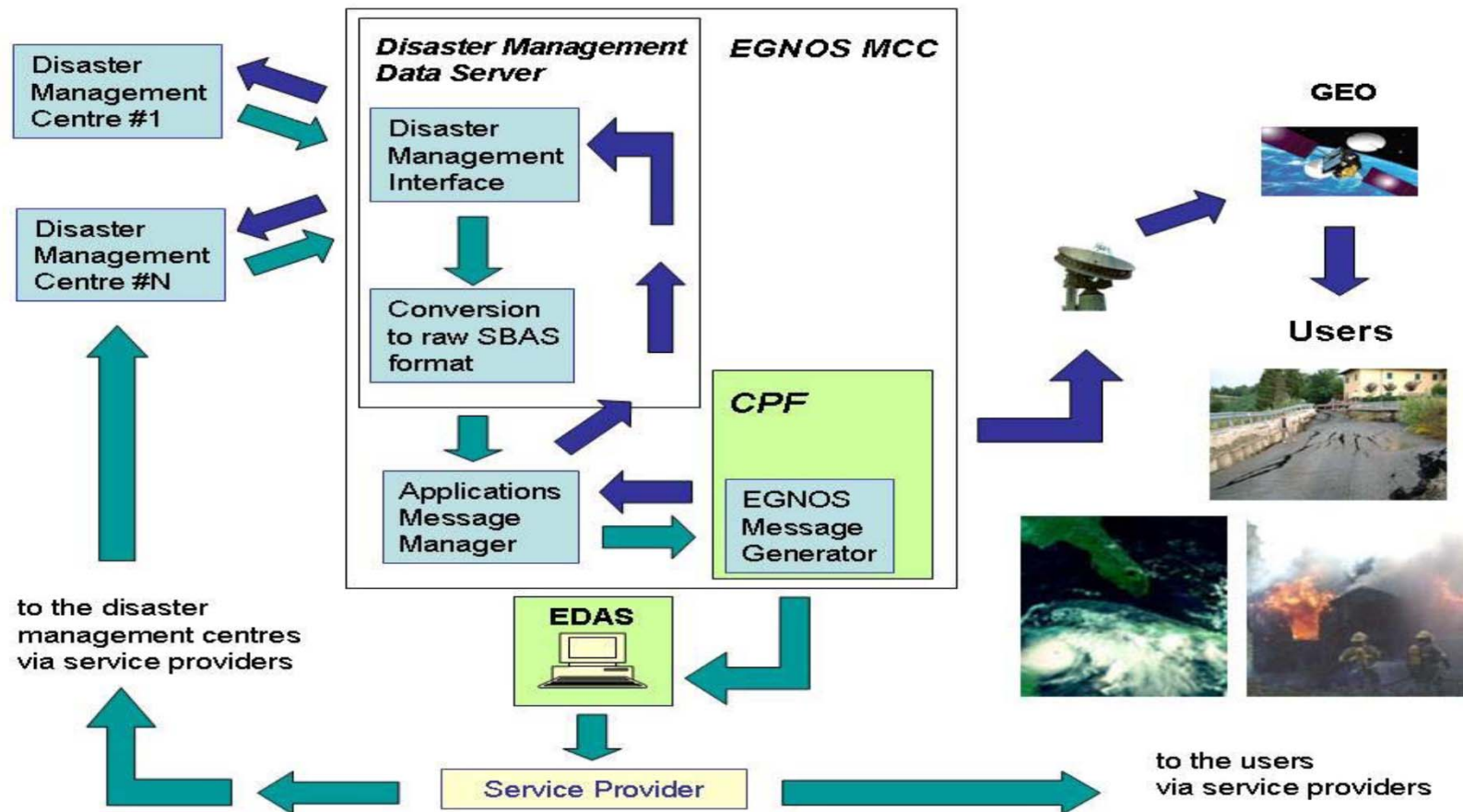


GPWS Display

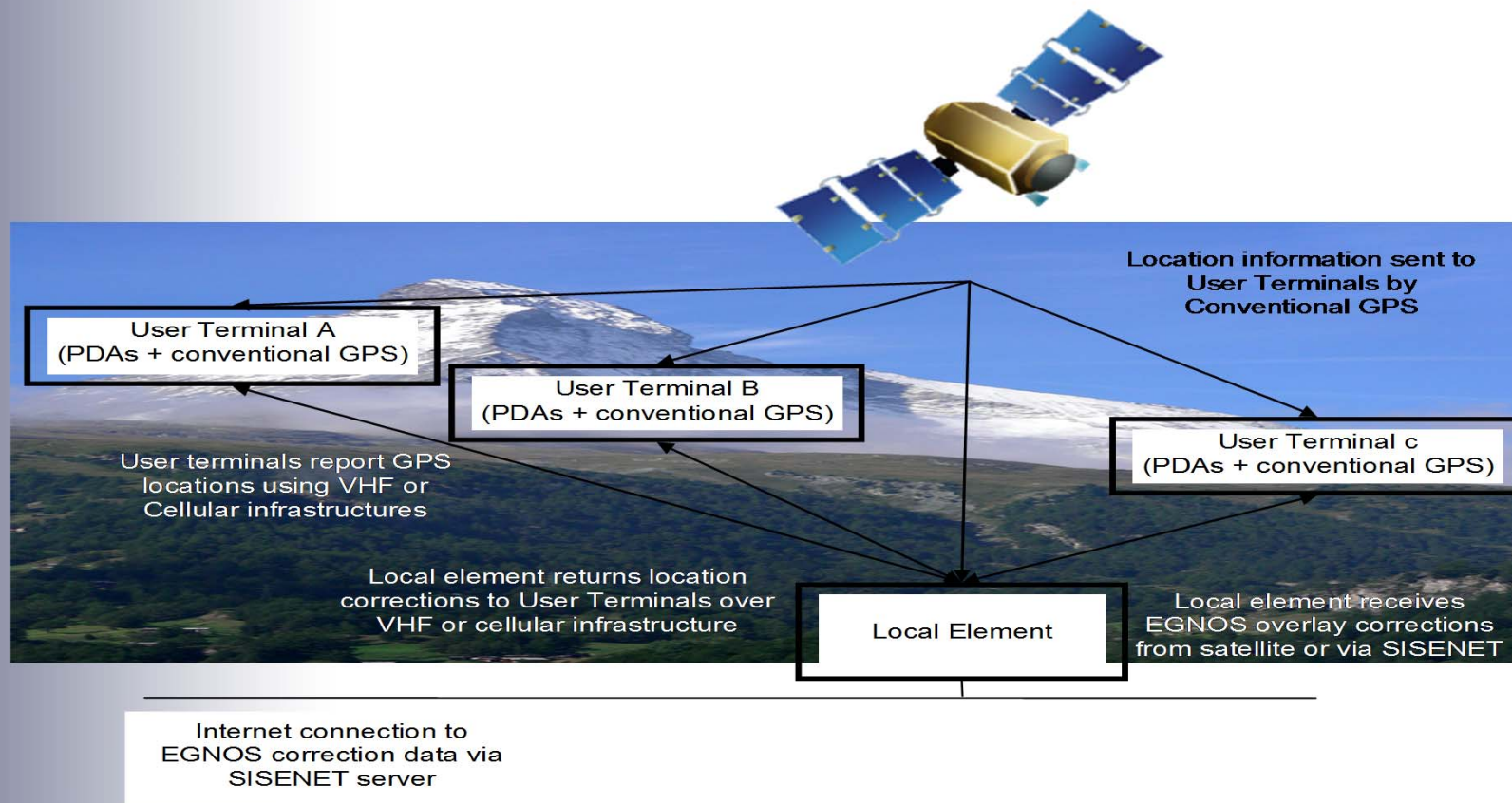
Crews must assimilate

- barometric and radio altitude instruments,
 - vertical speed indicator,
 - ground proximity warning systems,
 - terrain depiction systems,
 - flight management computer (FMC) etc.
- Opportunity for SBAS support...

New Forms of Interaction– ALIVE Architecture



Emergency broadcast if cellular infrastructure destroyed/unavailable.





New Forms of Interaction with SBAS



Integration of EC Projects in Emergency Infrastructures.

There Can Still Be Problems...

Current Architecture | EGNOS User Support - Windows Internet Explorer

http://egnos-user-support.essp-sas.eu/egnos_ops/egnos_system/system_description/current_architecture

File Edit View Favorites Tools Help

Current Architecture | EGNOS User Support

EGNOS User Support **ESSP**

Home > EGNOS SYSTEM > System description


SIGNAL IN SPACE

-  **NORMAL OPERATION**
- NORMAL OPERATION**
-  **NORMAL OPERATION**
- PRN124
-  **SIS OUTAGE**
Since: 07:55 UTC
- PRN126 (TEST)
Expected recovery: 19/05 07:50 UTC

Current Architecture

EGNOS is divided into four functional segments:

- 1) The ground segment is composed of the following stations/centres which are mainly distributed in Europe and are interconnected between themselves through a land network
 - 34 Ranging and Integrity Monitoring Stations (RIMS) + seven being deployed: receive the satellite signals and send this information to the MCC centres.
 - 4 MCC (control and processing centres) receive the information from the RIMS stations and generate correction messages to improve satellite signal accuracy and inform messages on the status of the satellites (integrity). The MCC acts as the EGNOS's 'brain'
 - GNILES (stations that access the geostationary satellites): they receive the correction messages from the CPFs for the upload of the data stream to the geostationary satellite and the generation of the GPS-like signal. This data is then transmitted to the EGNOS users via the geostationary satellite



EGNOS User Support

Home > EGNOS SYSTEM > System description

SIGNAL IN SPACE

-  **NORMAL OPERATION**
- PRN120
-  **NORMAL OPERATION**
- PRN124
-  **SIS OUTAGE**
Since: 07:55 UTC
- PRN126 (TEST)
Expected recovery: 19/05 07:50 UTC

- The processes used during
 - validation and verification.
- White and black boxes.
- Static and Dynamic techniques
- Case Study...

Any Questions...

