A Little Change…

…but a lot of work?

John Spriggs

SAF-SW Workshop & ES^2 Software Workshop,
EUROCONTROL IANS, Luxembourg, May 2013

NATS

(EC) No 482/2008 Applies to Changes

Article 3(1):

> *Whenever an organisation is required to implement a risk assessment and mitigation process in accordance with applicable Community or national law, it shall define and implement a software safety assurance system to deal specifically with EATMN software related aspects...*

Article 7:

> *It shall apply from 1 July 2010 to any changes to the software of EATMN systems…*

If the Regulation applies, you have to comply - however small the change.

(EC) No 482/2008 May Apply to Change

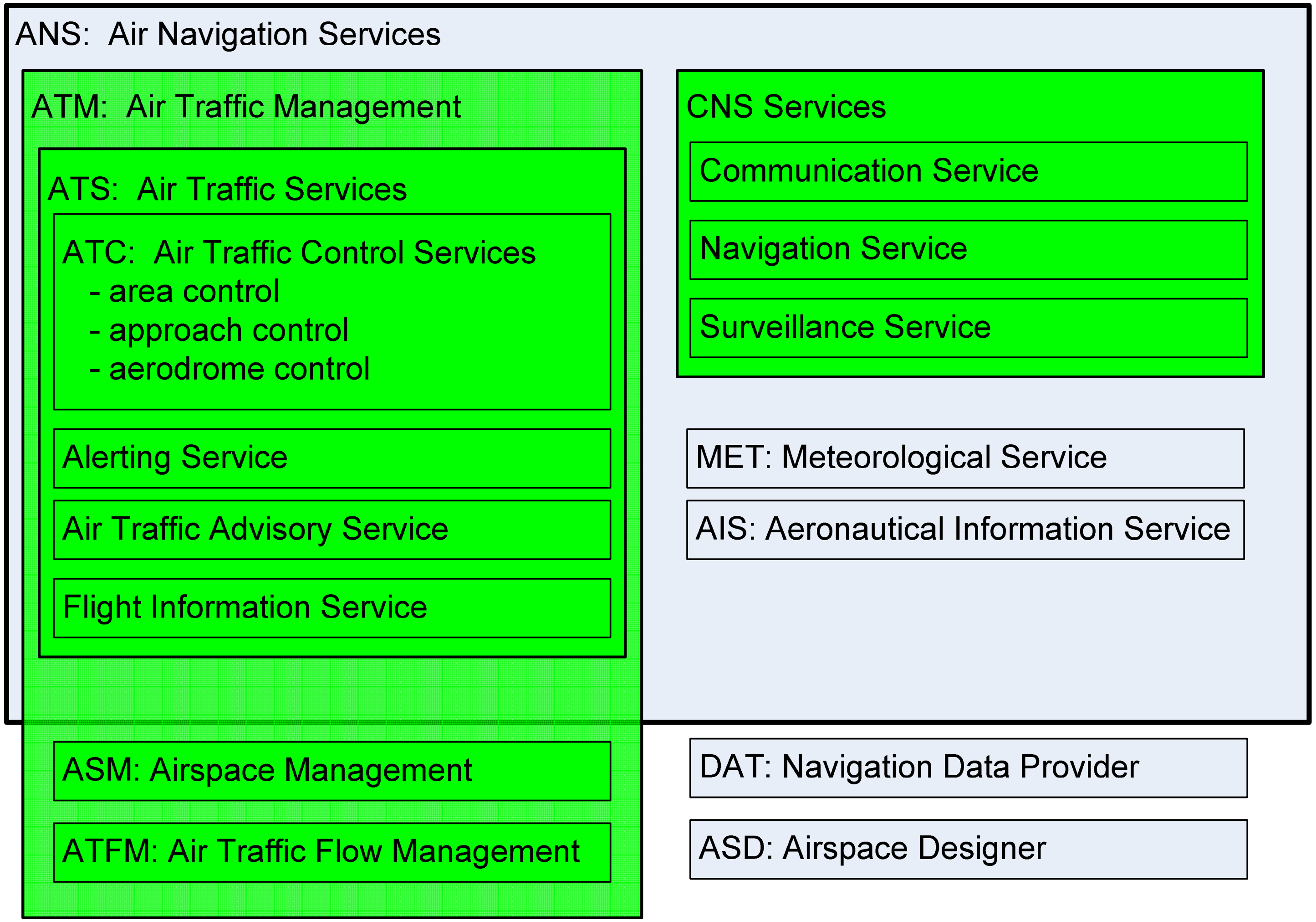
Wait! You said, “If the Regulation applies…”; are there times when it does not apply?

Article 1(2):

> *This Regulation shall apply to the new software and to any changes to the software of the systems for ATS, ASM, ATFM, and CNS.*

Are you making a change to a system that is **not** ATS, ASM, ATFM, or CNS? The Regulation does not apply, but there may be other legislative or regulatory requirements to take into account…

What Do All These Acronyms Mean?



ANS: Air Navigation Services

ATM: Air Traffic Management

ATS: Air Traffic Services

ATC: Air Traffic Control Services
- area control
- approach control
- aerodrome control

Alerting Service

Air Traffic Advisory Service

Flight Information Service

ASM: Airspace Management

ATFM: Air Traffic Flow Management

CNS Services

Communication Service

Navigation Service

Surveillance Service

MET: Meteorological Service

AIS: Aeronautical Information Service

DAT: Navigation Data Provider

ASD: Airspace Designer

However…

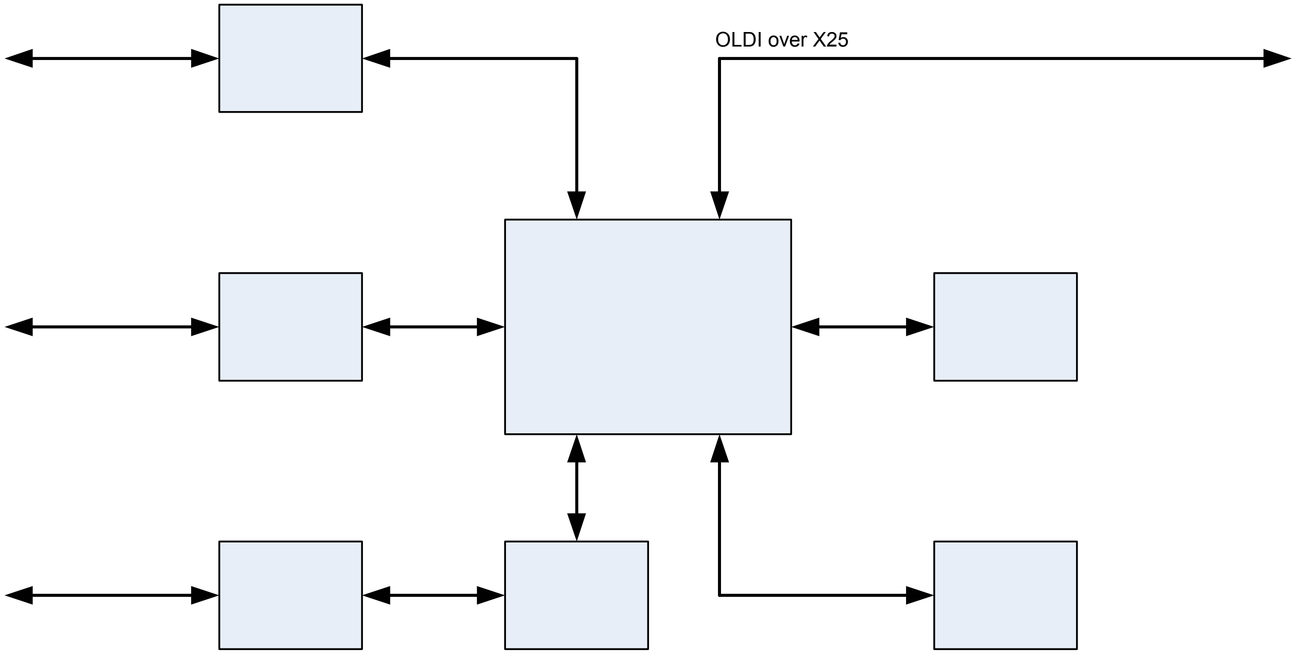
Do not assume that something is out of scope; get early agreement from your National Supervisory Authority.
You may need to provide a formal argument.

MET, for example, is the service providing meteorological data; the sub-system you use to receive those data may be considered, by the National Supervisory Authority, as part of an Air Traffic Services system, so may be in scope.

AIS - Aeronautical Information Services, for example, are out of the scope of (EC) No 482/2008, but the associated Implementing Rule, (EU) No 73/2010, has its own set of software assurance requirements…

Controlled Change

An existing system needs a new interface



Where do the effects of the change stop?

Controlled Change

Modify the Software...

Identify what is changed

Do you have documents with which to do this?

Identify what else the changes affect

Do the effects ripple out to other sub-systems?

Make the changes and provide the assurance

Are you sure it is enough?

...or Add a New Box?

Specify the box to have the same interface as before at the legacy side

Argue that the existing system is unaffected

Procure the box and provide the assurance, confident that the change is controlled

Could Some Changes Need Less Work?

Regulation (EC) No 482/2008 requires us to demonstrate various things to the National Supervisory Authority when we make a change:

- Requirements Validity
- Requirements Traceability
- “Safe” Functions
- Requirements Satisfaction
- Configuration Consistency

Do all these apply for all changes?

Requirements Validity

The Regulation requires us to demonstrate to the National Supervisory Authority that:

(a)the software safety requirements correctly state what is required by the software, in order to meet safety objectives and requirements, as identified by the risk assessment and mitigation process;

We are assuring a change; if there are no new software <u>safety</u> requirements, or if none are modified for the change, then it should be sufficient just to show that this is the case.

Requirements Traceability

The Regulation requires us to demonstrate to the National Supervisory Authority that:

> *(a) …;*
>
> *(b) traceability is addressed in respect of all software safety requirements;*

Again, if no software <u>safety</u> requirements are impacted by the change, or there are no new ones, it should be sufficient just to demonstrate this.

“Safe” Functions

The Regulation requires us to demonstrate to the National Supervisory Authority that:

> *(a) …;*
>
> *(b) …;*
>
> *(c) the software implementation contains no functions which adversely affect safety;*

“Functions which adversely affect safety” identified in risk assessment have safety requirements specified in mitigation. We need to assure that the requirements are met and that <u>no other functions interfere with correct operation of the mitigations.</u>

Requirements Satisfaction

The Regulation requires us to demonstrate to the National Supervisory Authority that:

(a) …;

(b) …;

(c) …;

(d) the EATMN software satisfies its requirements with a level of confidence which is consistent with the criticality of the software;

This follows from the last for safety requirements; we need to assure that they are met. But we always have to assure Requirements Satisfaction, as it is not limited to safety requirements.

Configuration Consistency

The Regulation also requires us to demonstrate to the National Supervisory Authority that:

> *(e) assurances are provided confirming that the general safety requirements set out in points (a) to (d) are satisfied, and the arguments that demonstrate the required assurances are at all times derived from:*
>
> > *(i) a known executable version of the software;*
> > *(ii) a known range of configuration data;*
> > *(iii) a known set of software products and descriptions, including specifications, that have been used in the production of that version.*

This one is essential; you must always do this, or what you provide will not be assurance…

Other Lessons Learned

- The software is changed, but there must be a bit that remains unchanged. Some of this will be affected by the changes, but some will be unaffected. We only need to assure the changed and affected parts, but run overall system tests to check that there are not unexpected effects…
- In particular, did you buy your software from a supplier with many Customers? How do you know that the change you requested has not been implemented with someone else’s change too?

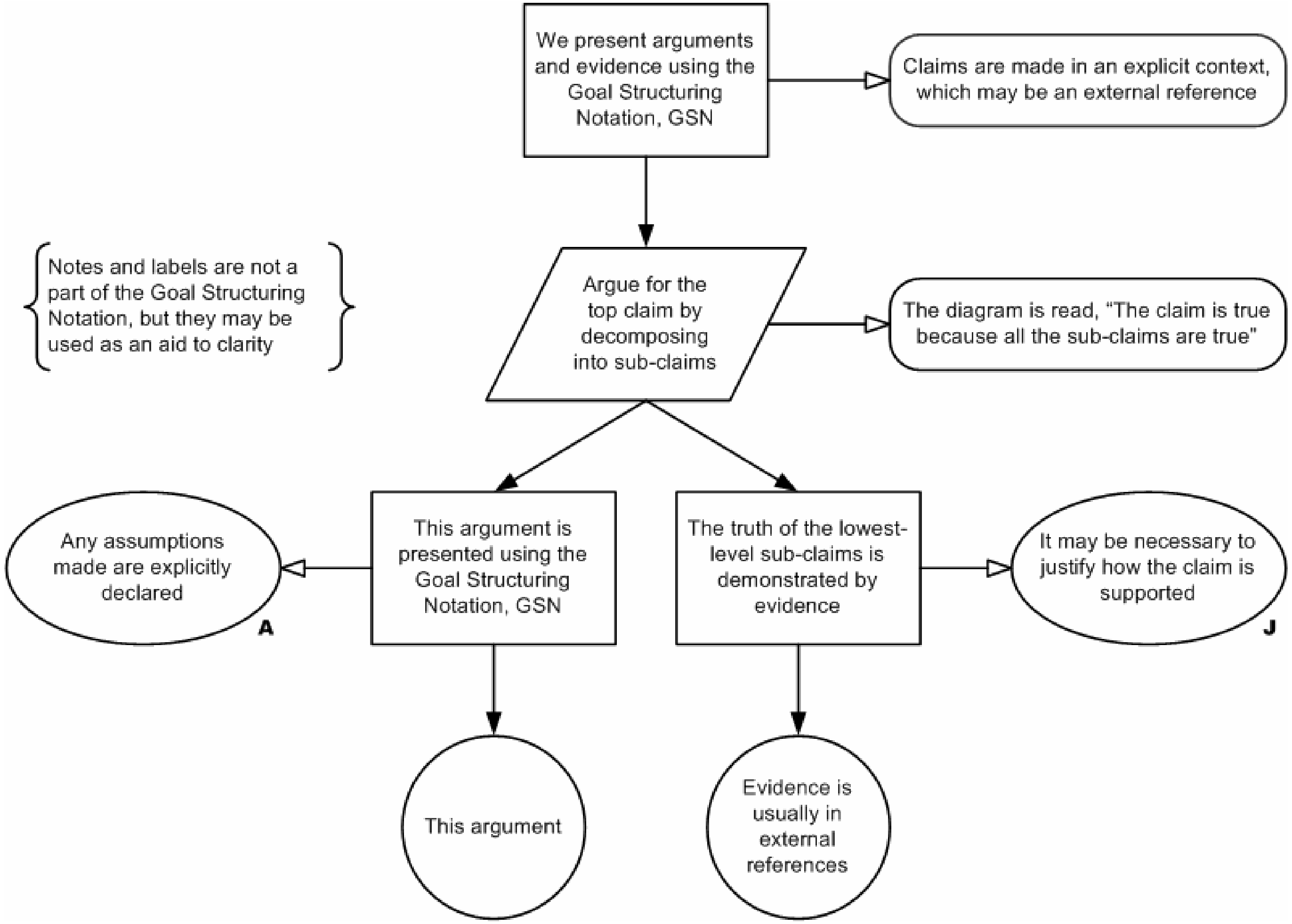
Other Lessons Learned

- When a change is too extensive or complex, it may be ‘best’ (time & €) to assure the whole thing as New software.

- Define your Assurance ‘Envelope’. Assure new software for a range of values of configuration data, for example.
  - ➢ As long as subsequent changes keep the software within this range, the assurance remains valid.
  - ➢ You need to argue that it is within the envelope.

- Present your assurance arguments and evidence using a graphical notation.

The Goal Structuring Notation



We present arguments and evidence using the Goal Structuring Notation, GSN

Claims are made in an explicit context, which may be an external reference

Notes and labels are not a part of the Goal Structuring Notation, but they may be used as an aid to clarity

Argue for the top claim by decomposing into sub-claims

The diagram is read, "The claim is true because all the sub-claims are true"

Any assumptions made are explicitly declared

This argument is presented using the Goal Structuring Notation, GSN

The truth of the lowest-level sub-claims is demonstrated by evidence

It may be necessary to justify how the claim is supported

This argument

Evidence is usually in external references

Summary of the Main Points

- Check whether your change is in scope of the Regulation; but also what other legislative or regulatory requirements may apply.

- Can you reduce the scope of the change by confining it to new software or equipment? …Or to an established assurance envelope?

- Does your National Supervisory Authority agree with your plans? How do they want you to present assurance? Get their agreement early in the project.

Do You Have Any Questions?