"The Ongoing Pursuit of an Integrated Toolset for Model-Based Design: Benefits and Challenges"

Steve Houghton
steven.d.houghton@selex-sas.com

SELEX
Sensors and Airborne Systems
A Finmeccanica Company

History

- **About 20 years ago I started work as a Software Engineer**

- **We took (textual) requirements allocated to Software and produced Software Requirement Specifications, Design Specifications, Test Specifications etc.**
  - Mostly in Word Perfect

- **We then progressed to taking (textual) requirements allocated to Software and produced similarly titled documents containing *Teamwork* diagrams.**

- **It felt the right thing to be doing**
  - **but** was virtually in isolation to every other engineering discipline

- **The only things that integrated our efforts were text documents and people**

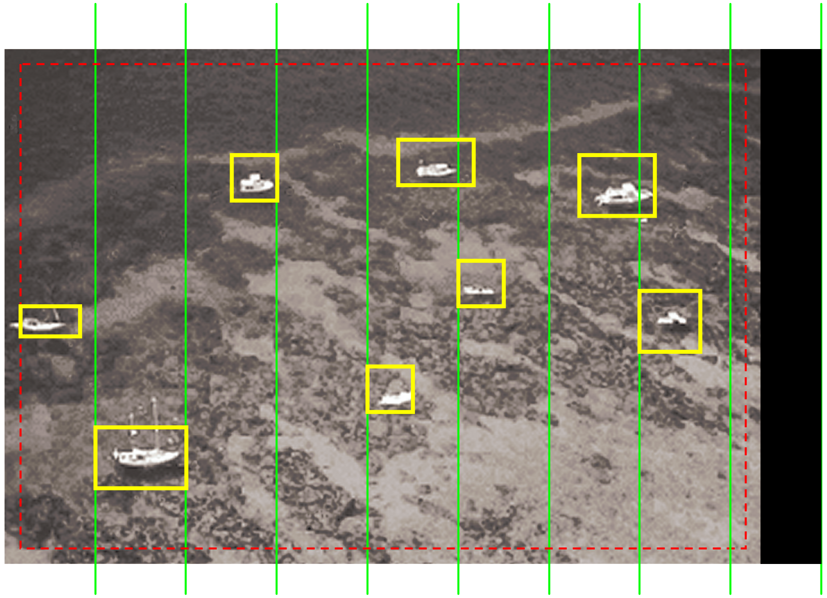
Subsequently…

- **Increase in product complexity and size**
- **Advances in Systems Engineering techniques**
    - Word Perfect wasn't perfect after all
- **Advances in Software Engineering techniques**
    - UML, auto-code generation
- **Corresponding advances in tools to support these techniques**
    - System modelling tools
    - Mathematical modelling tools
    - Auto-code generations tools

A new way

- **SELEX S&AS have undertaken projects using and attempting to integrate these new tools**
- **I’ll briefly highlight a couple of these today:-**
  - An Airborne target recognition system using a Burst Illumination Laser (BIL). This used a MathWorks / XILINX/ Nallatech toolset for algorithm development and subsequent deployment to an FPGA.
  - The Sea Wolf Missile Upgrade programme (SWMLU) which uses a MathWorks/ Simulink/ Rhapsody toolset.
- **I’ll then discuss the benefits & challenges we experienced**

BIL – Overview and Requirement



Target Detection Capability

- Processing Platform (PC104+ rig)
- Function calls (software) for host CPU
- Enhanced Algorithm

Available time to complete task – 6 months

(est. time using conv processes – 12 to 18 mths)

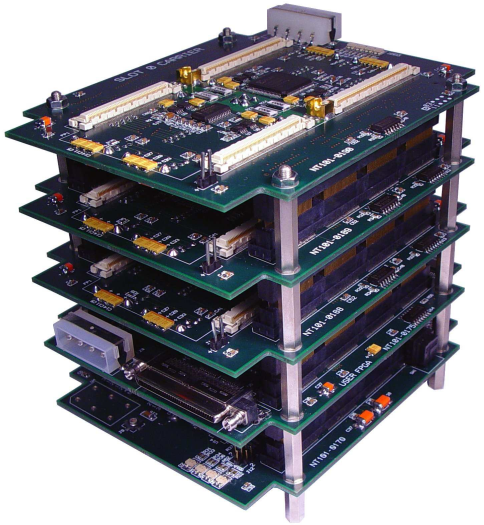
Team - 1 HW, 1 Sys & 1 p/t SW

Performance Requirements -

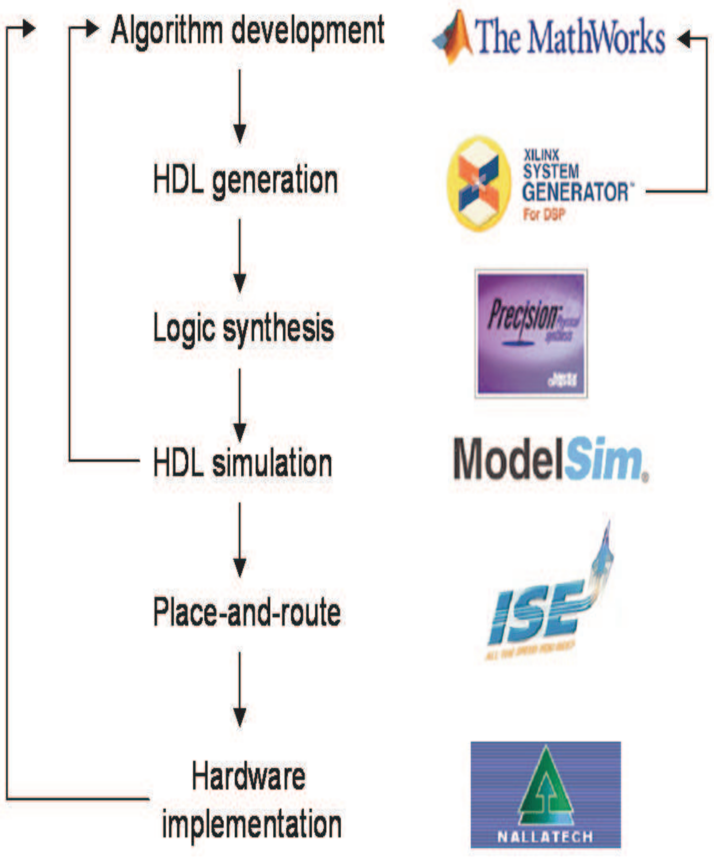
- Processing Time < 1s for 11M Pixel

(& Programmable Image Size Capability)

- Detection Rate & False Alarm Rate specified



BIL - SELEX Model Driven Engineering Process



Maths to Hardware Seamlessly

Infrastructure facilitates Model Driven Philosophy

- Model is FPGA Requirement Spec
- Verification through Feedback
- Solid foundations (re-use of existing toolkits/experience)
- Promotes cross-functional awareness & teamwork

plus

Verification Options
Enhanced Verification Capabilities

BIL - The Deliverable & Performance Metrics

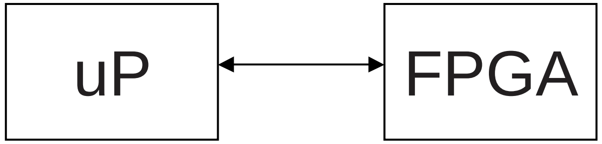
Deliverables

- COTS FPGA based module

(as Image Processing Platform)

- Function calls (embedded in host CPU)
- New algorithm (targeted to FPGA)

	Detection > 90% (per image averaged over test imagery)	False Alarm Rate > 1% (per image averaged over test imagery)
Customer	**73%**	**74%**
SELEX-SAS	**91%**	**5%**



Outcomes

- Requirements Exceeded
- Processing time 0.41s
- On time, on budget
- Delighted customer

SWMLU - Overview and Requirement

Sea **W**olf **M**id **L**ife **U**pdate programme

Uses data from EO system and radar system to identify, track and prioritise threats

The System level requirements are captured in DOORS

Simulink and Rhapsody have been used by developers in fulfilling systems requirements

The development has required 5 algorithmic modellers and 8 software developers

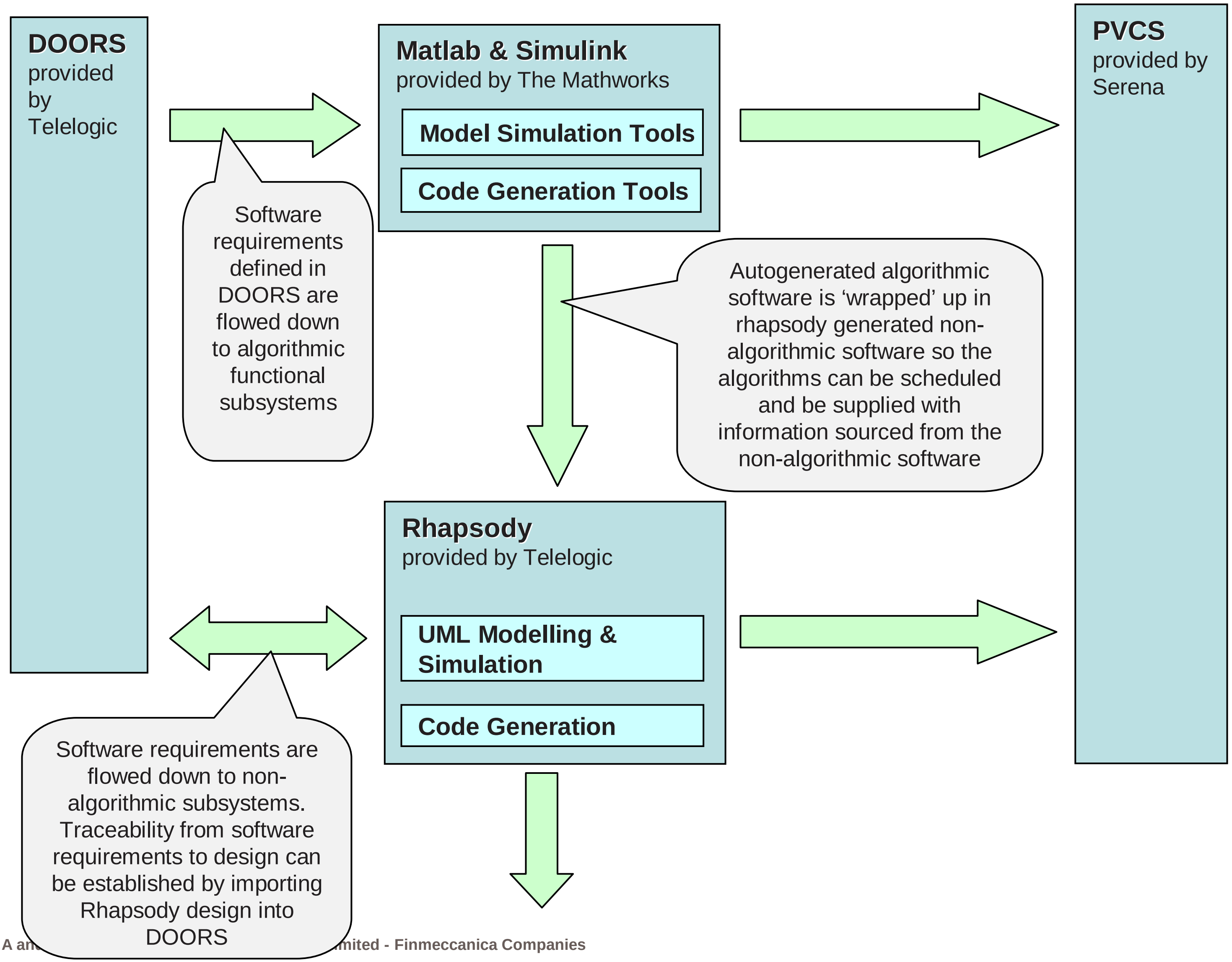
The development has produced approximately 560K lines of code

- 40% Simulink generated
- 60% Rhapsody generated

Project is now in the design proving phase

SWMLU - The Process

SELEX
Sensors and Airborne Systems
A Finmeccanica Company



DOORS
provided by Telelogic

Matlab & Simulink
provided by The Mathworks

Model Simulation Tools

Code Generation Tools

PVCS
provided by Serena

Software requirements defined in DOORS are flowed down to algorithmic functional subsystems

Autogenerated algorithmic software is ‘wrapped’ up in rhapsody generated non-algorithmic software so the algorithms can be scheduled and be supplied with information sourced from the non-algorithmic software

Rhapsody
provided by Telelogic

UML Modelling & Simulation

Code Generation

Software requirements are flowed down to non-algorithmic subsystems. Traceability from software requirements to design can be established by importing Rhapsody design into DOORS

“Roll out” strategy - Workshops

- **Tool introduction / demonstration to**
  - Software and FPGA Engineers
  - Systems Engineers
  - Chief Engineers
  - Project Managers
  - … order should have been reversed!

- **Lunchtime Learners**
  - Variety of nominated and prioritised technical topics

- **Internal technical presentation of project work**

“Roll out” strategy

- **Training**
  - Standard courses good for fundamentals
  - Custom courses developed to address specific skill gaps
  - 1:1 mentoring

- **Consultancy**
  - Highly effective where applied
  - But most teams take ‘encouragement’ to accept external help

- **Review**
  - Looking for opportunities for tool deployment
  - ‘User’ interviews

- **Knowledge sharing**
  - Creation of user communities

“Challenges” experienced – 1

- **Systems Modellers produced models that the Engineers couldn’t implement**
  - The wall was down but they were initially throwing the bricks at each other
  - Brought long standing barriers to the forefront but where solutions could be developed
  - “Pairing up” solution

- **Initial Rhapsody/Simulink integration took months**
  - Last entire rebuild/integration took 3 days!

- **IT implementation issues**
  - Virus checker
  - Usage data

- **Risk adversity**
  - Project Managers opposing new technologies
  - Engineers need to be confident in the toolset and process

- **Didn’t fit with our existing Design/Phase review process particularly well**

“Challenges” experienced - 2

- **Suitability of ‘pilot’ projects**
  - ‘In at the deep end’ isn’t *necessarily* the best philosophy
  - Pressure to deliver savings on the first project(s)

- **Producing C++ code**
  - Was poorly supported by debug tools (better now)

- **Timing of training**
  - JIT difficult to deliver
  - Too late – missed opportunity
  - Too soon – knowledge retention problem

- **Developing highly marketable engineers**
  - Need to manage staff attrition rates
  - Is also a positive

Key Benefits / Achievements

- **Common Toolset – true synergy between disciplines**
  - Increased cross-functional awareness
  - The Systems Modellers work with the Engineers (SW and HW)
  - Less waste, increased productivity, quicker to market
- **System proven to 'work' before implementation**
  - Early models delivered to customer
- **Reuse**
  - Significant reuse has occurred of sub-models
- **Complete process developed for future projects**
- **UML (Rhapsody) and Simulink can be applied together easily and effectively**
- **User communities implemented at both site and group levels**

What next?

- New projects will use Model Based Design
- All systems modellers and software engineers will have been trained in tools by mid 2008
- User Communities to facilitate knowledge sharing across organisation
- Increased use of consultancy to ensure effectiveness/success
- Continue process development

Recommendations

- Look for and resolve any IT issues
- Tool exposure *before* attending training courses
- Plan and mandate the use of consultancy
- Assign responsibility for model integration to a team of systems modellers and software/hardware engineers
- Capture and review experiences to refine process
- Develop a ‘coding standard’ / style for models
- Use the right tool for the right job. No single tool does it all (well)
- Support the development of User Communities

Thank you

Steve Houghton
steven.d.houghton@selex-sas.com

SELEX
Sensors and Airborne Systems
A Finmeccanica Company