DEFENCE R&D DÉFENSE

FOSS in Military Computing

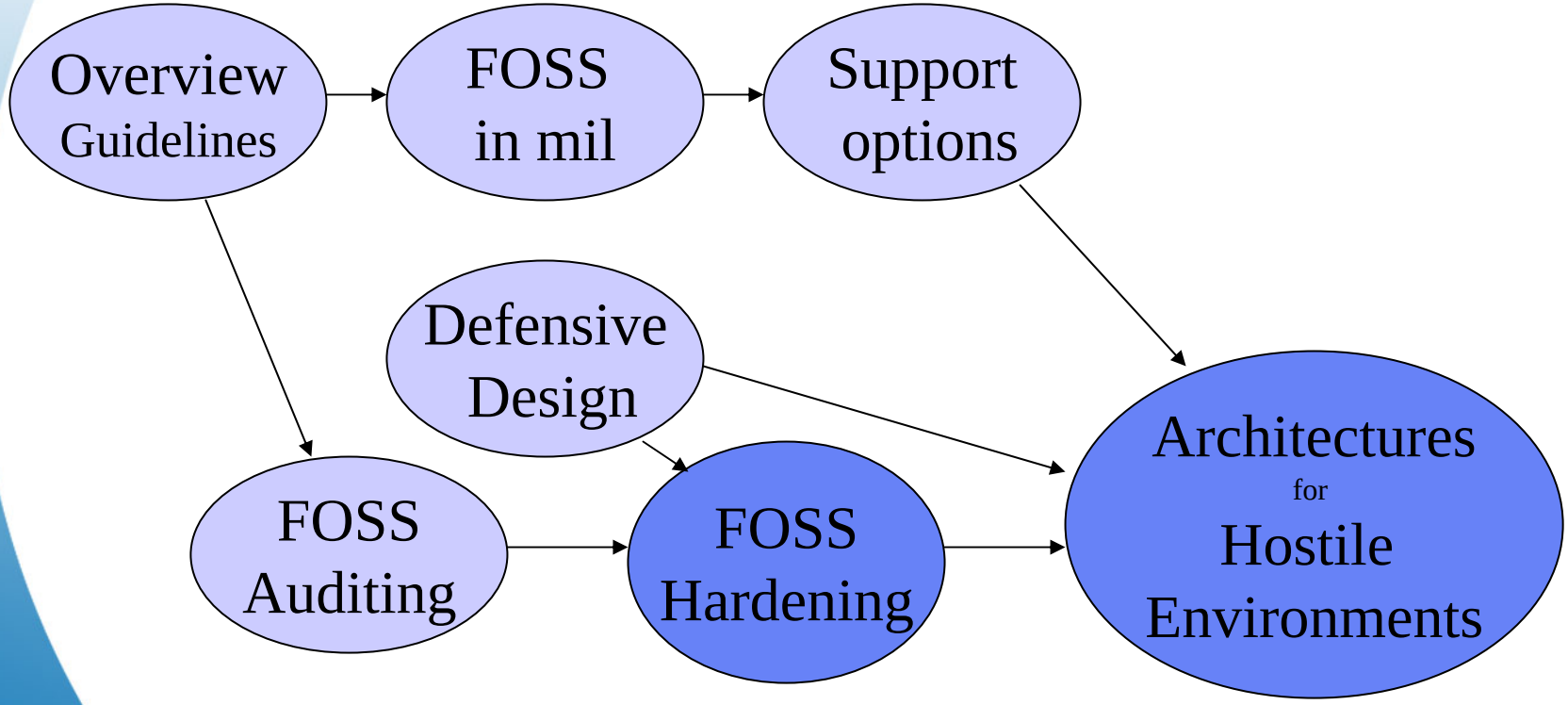
Life-Cycle Support for FOSS-Based Information Systems

By

Robert Charpentier

Richard Carbone

FOSS Project History



FOSS: Free and Open Source Software

Contents

- TTCP JSA Workshop on FOSS in Military Computing
- Canadian Navy's Context
- Strategy Recommended to the Canadian Navy
- Basic Support Models Applicable to FOSS
- Trends and Recommendations

TTCP: The Technical Cooperation Program
JSA: Joint Systems and Analysis
FOSS: Free and Open Source Software

JSA Workshop Participants

Workshop Key Findings

- TTCP nations are all experimenting with FOSS
- FOSS is a useful technology for military computing
- Use of FOSS should be increasing in military computing
- R&D communities should demonstrate leadership
- Open source can prevent loss of research

TTCP: The Technical Cooperation Program
FOSS: Free and Open Source Software

Lessons Learned and Best Practices with FOSS

- Adoption of Open Standards should be prioritized
- FOSS must be selected on its technical advantages
- Hybrid architectures using COTS and FOSS are often best
- Access to source code has proven very beneficial and practical
- Configuration control is required to comply with licensing constraints

COTS: Commercial-Off-The-Shelf
FOSS: Free and Open Source Software

Pending Issues Deserving More Attention

- The strategic value of FOSS is still inadequately understood
- Adjustments may be needed in military organisations to migrate to FOSS
- Cost modeling is clearly inadequate and may be impracticable
- License interpretation is complex and requires immediate attention
- Understanding potentially offensive FOSS is critical

Life-Cycle Support
for
FOSS-Based Information Systems

Canadian Navy's Context

- Modernized C2 Systems on Frigates
  - *Halifax Class*
  - *Life expectancy → circa 2025*
- Open Architecture philosophy
  - *Open standards - Prioritized*
  - *Leading to FOSS conformant to Open Standards*
- Case study: *Linux as basic Operating System on ships*
  - *Best life cycle options for Linux in Canadian Navy*

C2: Command and Control
FOSS: Open Source Software

Study Key Findings

- FOSS offers more options than COTS software in terms of long-term maintenance
  - Four basic support models
  - Very strong complementarities of the different options
- Support Models to be assessed and validated
  - *In each specific context*
  - *On a case by case basis*

COTS: Commercial-Off-The-Shelf
FOSS: Free and Open Source Software

Strategy Recommended to our Navy

- First, try vendor support (as done with COTS)
- Alternatively, seek consultant support
- Then, consider joining a consortium
- As a last resort, take full responsibility of the software

Vendor Support

- Buy support from FOSS reseller/integrator as done with COTS Software
- Pro: Known model
  - Low demand on internal resources
  - Efficient leveraging of user experiences (and in debugging)
  - Available training , simpler accountability and easier license management
- Con: Little influence on product evolution
  - Dependence on suppliers (lock-in)
  - Little support available for less popular FOSS
- List of integrators offering support included in the report
- Author's assessment:

  Ideal to reduce anxiety associated with FOSS adoption

Consultant Support

- Buy support from consultants (like done with some CF software)
- Pro: Known model
  - Moderate demand on internal resources
  - Some significant influence on product evolution
  - Can be optimized for specialized requirements (e.g. classified systems)
- Con: Little leveraging of user experiences and debugging
  - Potentially very expensive and legally complex
  - Dependence on the availability (and maintenance) of competent resources
- List of major consultant firms offering support in appendix
- Author's assessment:

  Could be ideal for specific requirements (rather limited in scope) but, probably too expensive as a prime source of support

Support From a Consortium

- Join a consortium to exchange software enhancements
- Pro: Tremendous offering from the FOSS communities
  - Medium demand on internal resources
  - Very efficient leveraging of user's experiences and debugging
- Con: New paradigm to understand and to integrate
  - Contributions will be expected, including improved source code
  - Free support services typically bear no legal responsibilities
  - Documentation may be less user-friendly
- List of consortia offering support included in the report
- Author's assessment:

  Probably cost effective but demands significant changes in current practices

Internal Support

- Taking responsibility for the maintenance and the evolution of the FOSS
- Pro: Full control on the technology
  - Can be extended beyond economic viability of a product
  - Money spent on local resources in building in-house expertise
  - Protection against automated attacks via code diversity and reduced packages
- Con: High usage of internal resources to develop and maintain
  - Expensive (no leveraging)
  - Longer development delays
  - All legal responsibilities to be taken on internally
- Author's assessment:

  The most demanding option -- financially and legally!

Basic Approach in Ranking Support Options

- Define well your maintenance context
- Review and prioritize criteria applicable to support plan
- Consider all appropriate support options
- Review cost estimates
- Prototype the selected FOSS-based architecture
- Elaborate a contingency plan before deployment
- Seek approval and document lessons learned

Trends and Recommendations

- Guidelines for FOSS support needed now
  - *FOSS already in use in DND/CF*
  - *Expansion predicted by all tech forecasting firms*
- Training and some experimentation is recommended
- FOSS-based OS can be supported by C2IS developers
  - *Rather than hardware manufacturer*
  - *Faster response to cyber-threat*

Concluding Remarks

- Openness is mandatory to build mutual trust
- Security must now evolve from patterns to standards
- Security not only in development

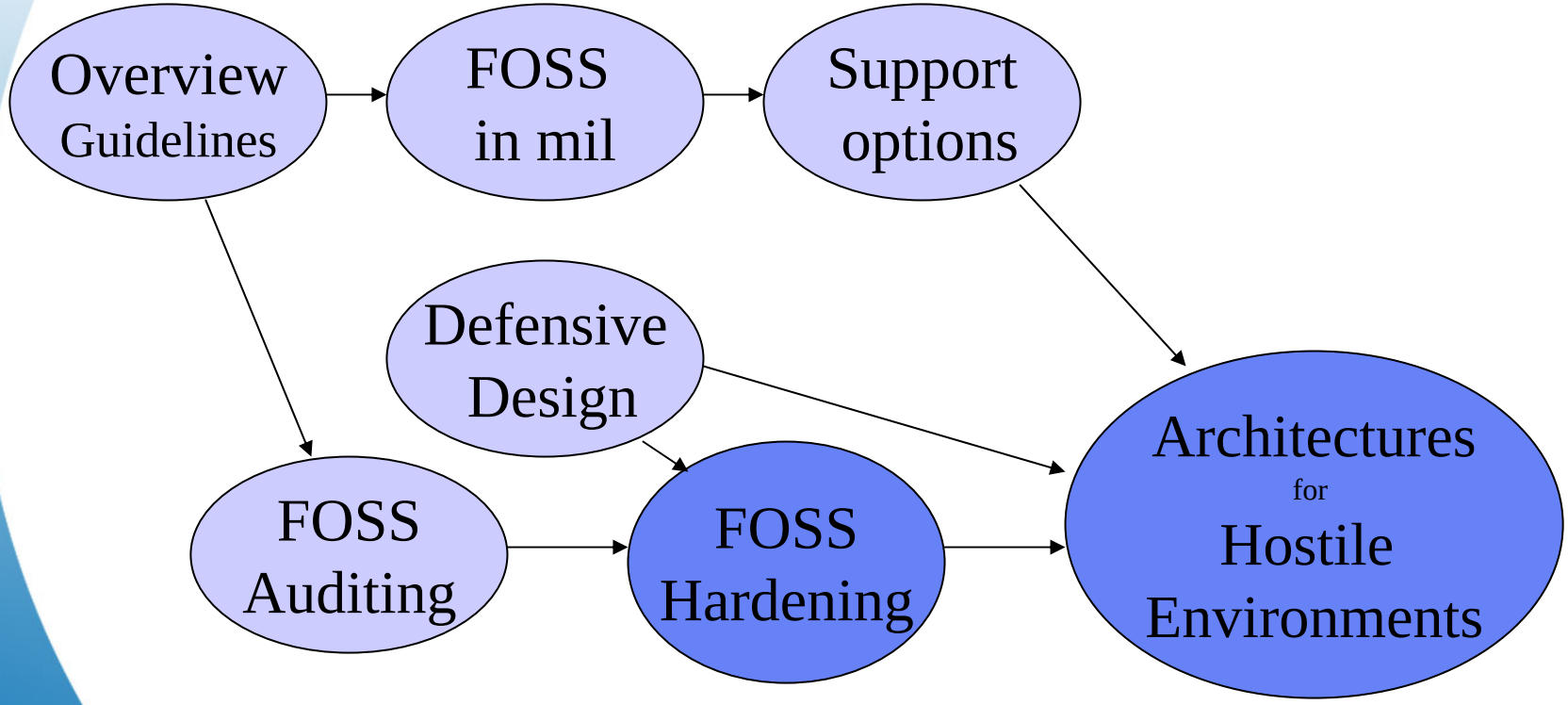
  but also part of life-cycle maintenance
- Migration to modern languages must be prioritized

For comments :

FOSS@drdc-rddc.gc.ca

DEFENCE R&D DÉFENSE

FOSS Project History



FOSS: Free and Open Source Software