Software Engineering for Fusion Reactor Design

Presented by Wayne Arter
CCFE, Culham Science Centre, Abingdon, Oxon., UK
Show how we have produced software capable of further development after 30+yr:

1. Background, the ITER experiment.
2. History of scientific software tools at Culham.
3. Designing one piece of software for 30+ years of development and use by ITER.
4. Producing a more comprehensive reactor design tool.
- World’s “fusion experiment” (magnetic confinement)
- Tokamak (magnetic doughnut)
World’s “fusion experiment”
Cadrache, S. France

www.iter.org
Towards Electricity Generation

<table>
<thead>
<tr>
<th>Year</th>
<th>0</th>
<th>10</th>
<th>20</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITER</td>
<td>Construct</td>
<td>Operate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IFMIF</td>
<td>Design</td>
<td>Construct</td>
<td>Operate</td>
<td></td>
</tr>
<tr>
<td>DEMO</td>
<td>Explore</td>
<td>Concept</td>
<td>R&amp;D</td>
<td>Design</td>
</tr>
</tbody>
</table>

**JET**

- $80 \text{ m}^3$
- $\sim 16 \text{ MW}_{th}$

**ITER**

- $800 \text{ m}^3$
- $\sim 500 \text{ MW}_{th}$

**DEMO**

- $\sim 1000 - 3500 \text{ m}^3$
- $\sim 2000 - 4000 \text{ MW}_{th}$

iter.org ccfe.ac.uk
History of scientific software tools at Culham
Leading players

▪ K.V. Roberts (to 1983)
  ▪ 1969 established journal Computer Physics Communications (CPC, Impact factor 3.9)
  ▪ CPC Program Library
  ▪ “Software should be readable like a book” [2]

▪ J.W. Eastwood (to 1996)
  ▪ Further developed OLYMPUS programming system, became U(nix)-OLYMPUS c.1990
  ▪ “Computer Simulation Using Particles” with Roger W. Hockney, P³M algorithm, 3DPIC/EMX
U-OLYMPUS codes from mid-1980s are still capable not only of use but also of *development*.

- Basically strict FORTRAN 66 standard with minimal extensions for character handling.
- *Same language also* for preprocessing software, for code generation and formatting.
- *Same language* for graphics (Culham GHOST).
- C-shell wrapper, including makefile, version control via SCCS.
- Descriptive publications, man pages in troff/nroff
- New Fortran standards address many of deficiencies met by U-OLYMPUS
- Insufficient resource/gain to update U-OLYMPUS
- Concentration on programming style after Brealey, now object-oriented, see ref. [1] of abstract, ie. Report CCFE-R(15)34, and templates at https://github.com/wayne-arter/smardda-qprog.git
- C shell -> Bash shell utilities, including git for version control and general provenancing.
My contribution to ITER
- Confinement of plasma in tokamaks is not perfect and MegaWatts (MW) of energy expected to leak from outer midplane region – both projects to model “where power goes”
- Simple physics that power flows along fieldlines to “first” wall/plasma facing components (PFCs).
- Magnetic field-line tracing over complex geometry
- Up to $10^6$ field-lines each as up to $10^5$ rays using hybrid SMART/DDA algorithm (SMARDDA)
• 21st Century development, started 2008.
• Modular object-oriented software based around the SMARDDA ray-tracing algorithm for triangulated surface geometries
• Written to own published Fortran-95 software standard [1]
• Documented in 2 publications
• Originally for neutronics and neutral beam duct design, adapted for ITER in SMITER project, now being used for reactor design, coupled to COSSAN sensitivity analysis/UQ software
• Linux/Mac ifort and gfortran + bash 3.0

2 372 343 triangles
Applications of SMARDDA Modules

IFMIF – neutron source

Neutral Beam Ducts - reionisation

ITER

MAST-U
Divertors and Limiters

ITER
Power deposition on PFCs

JET NB source

DEMO

FTU-LLL
SMITER (SMARDDA for ITER)

Specifications from customer

- User-friendly interface with GUI
- (Integration into ITER code-base IMAS)
- Capabilities for
  - Verify existing designs as physics basis improves
  - Interpret data from experiments (2025 on)
  - Integrate into real-time control system
- CCFE constraint that SMARDDA modules remain usable for commercial fusion reactor design, i.e. capable of integration into suite of engineering tools such as ANSYS.
History implied:

- Use of object-oriented Fortran 95 programming style
- Bash 3.0 wrapper including makefile
- Semi-personal awk/sed/grep development tools (not part of deliverable) for code generation

Additionally:

3-D surface meshes as legacy .vtk, from CATIA™ output of NASTRAN™ .dat geometry files. Implies visualisation by ParaView (and gnuplot).
- Set of Fortran-95 objects orchestrated by bash 3.0 script
- Unix dialog used for GUI, “training only”
Separate git repo for Fortran/bash source and for data for each project

- Project Script
- COSSAN UQ tool
- Run Script (single case)
- Cmdwrap (logging)
- Fortran-95 executable

Cmdwrap directory records time, executable and data
Reactor Design Problem
Aims (SMITER project as microcosm):

- Predict, interpret and control operation of burning plasma experiment such as ITER.
- Design demonstration fusion reactor such as DEMO.
• Same issues as for any electrical power station:
  a. Fluid flow, heat transfer
  b. Mechanical stress, etc. etc.
• Large uncertainties in reactor core and nuclear data:
  • Plasma turbulence affecting
    a. Fusion yield (exponentially on T - heat transport)
    b. Power exhaust (heat leaks out to first wall)
    c. Instability (intermittency, e.g. ELMs)
  • Nuclear data cross-sections
  • Material properties under irradiation by 14MeV neutrons
Software Progression

a) Power balance model (Zero-D) such as in CCFE’s PROCESS code (constrained optimiser) 

----------develop----------

b) +

c) Split into sub-problems for more detailed 1-, 2-, …, 6-D modelling and define inter-relationships

d) Define objects within sub-problems

And recurse and refine over 30+ years

21 SEA - April 2018
Guidelines for Development

• Exploit expertise of different professionals.
• Ensure software always produces an answer, always with error estimates.
• Ensure software capable of spectral accuracy.
• Redundancy in major components.
• Recommend rather than impose standards, software tools and libraries.
• Have a policy on use of OpenSource software
<table>
<thead>
<tr>
<th>Professional</th>
<th>Main role</th>
<th>Main code usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engineer</td>
<td>Design</td>
<td>Ensembles/Scans</td>
</tr>
<tr>
<td>Physicist</td>
<td>Select and define physics</td>
<td>Single run</td>
</tr>
<tr>
<td>Mathematician</td>
<td>Solvers and algorithms</td>
<td>Library/subprogram</td>
</tr>
<tr>
<td>Computer Scientist</td>
<td>Compilation and hardware</td>
<td>Machine Level</td>
</tr>
</tbody>
</table>

- Blurring of roles is expected
<table>
<thead>
<tr>
<th>Professional</th>
<th>Main role</th>
<th>Main code usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engineer (Arter)</td>
<td>Design</td>
<td>Ensembles/Scans</td>
</tr>
<tr>
<td>Physicist (Eastwood)</td>
<td>Select and define physics</td>
<td>Single run</td>
</tr>
<tr>
<td>Mathematician (Brealey)</td>
<td>Solvers and algorithms</td>
<td>Library/subprogram</td>
</tr>
<tr>
<td>Computer Scientist (Hockney)</td>
<td>Compilation and hardware</td>
<td>Machine Level</td>
</tr>
</tbody>
</table>

- Blurring of roles is expected *(3DPIC/EMX development)*
- Only subset of interrelationships shown
- Feedback from right to left to complete control loop
- Feedbacks and physics of energy transport demand close-coupling, via one matrix equation.
- Robustness and flexibility point to object-oriented design.
- Objects whatever their meaning must ultimately define matrix coefficients.
- Coefficient definition indirect through other matrix, particle or ray-tracing calculations.
Summary

- Outlined the key modelling issues for design of fusion reactor core:
  - SMARDDA software as the microcosm.
  - Need for software capable of continuous development.
- History at Culham implies that it is possible to design software for 30 years of development.
- Guidelines for selecting a solution presented.
- Possible solution outlined.
Acknowledgements

This work has been partly carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053 and from the RCUK Energy Programme[grant number EP/P012450/1]. Other work reported was funded by F4E under project F4E-OPE-148 and ITER under contract ITER/RT/14/4300000958.


