

The MESH Modelling Community

latest developments

Bruce Davison, Brenda Toth, Dan Magosse, Chris White
Meteorological Service of Canada
Hydrometeorology and Arctic Lab,
National Hydrology Research Centre
Environment Canada
Saskatoon, Saskatchewan

Vincent Fortin, Diana Versegny, Ric Soulis, Frank Seglenieks, Alain Pietroniro, Pierre Pellerin, Saul Marin, Edgar Herrera, Matt MacDonald, Robert Armstrong, Pablo Dornes, Isabelle Doré, Marco Carrera



Outline

Management rather than Science

- **Vision of Community Modelling**
- **Software Engineering**
- **Status of our work**

Vision of Community Modelling

Monitoring – Research – Development – Operations – Decision Support

Numerical Weather Prediction

Climate Prediction

Hydrologic Modelling

Ecological Modelling

Groundwater Modelling



This already exists!

(But there is room for improvement.)

Weaknesses in the System

Monitoring – Research – Development – Operations – Decision Support

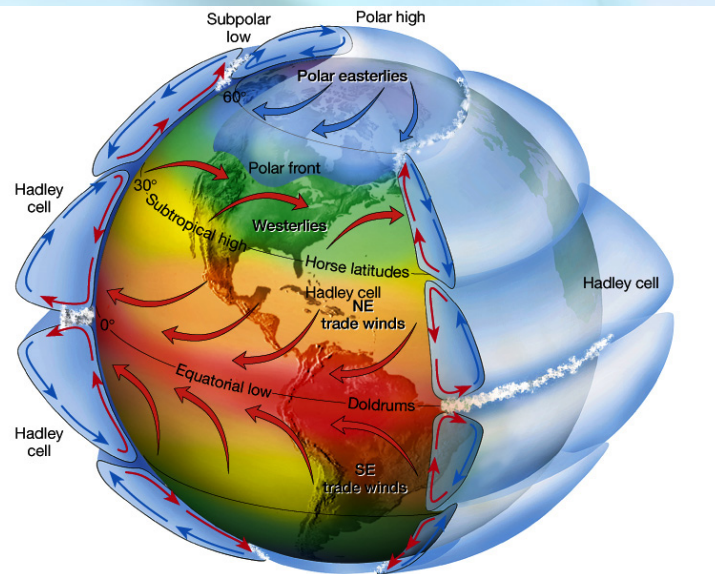
Numerical Weather Prediction

Climate Prediction

Hydrologic Modelling

Ecological Modelling

Groundwater Modelling



- Weak links
- Little emphasis on development

Our Goal – Focused Improvements

Monitoring – Research – Development – Operations – Decision Support

- Weak links – improve the links
- Little emphasis on development – improve development practices

Leads to...

Software Engineering

What is Software Engineering?

Table 1 The SWEBOK Knowledge Areas (KAs)

- **Software requirements**
- **Software design**
- **Software construction**
- **Software testing**
- **Software maintenance**
- **Software configuration management**
- **Software engineering management**
- **Software engineering process**
- **Software engineering tools and methods**
- **Software quality**

Table 2 Related disciplines

- **Computer engineering**
- **Computer science**
- **Management**
- **Mathematics**
- **Project Management**
- **Quality Management**
- **Software Ergonomics**
- **Systems Engineering**

What is Software Engineering?

Table 1 The SWEBOK Knowledge Areas (KAs)

- Software requirements
- Software design
- **Software construction**
- Software testing
- Software maintenance
- **Software configuration management**
- **Software engineering management**
- Software engineering process
- **Software engineering tools and methods**
- Software quality

Table 2 Related disciplines

- Computer engineering
- Computer science
- Management
- Mathematics
- Project Management
- Quality Management
- Software Ergonomics
- Systems Engineering

What are the knowledge areas?

1. The Software Requirements Knowledge Area (KA) is concerned with the elicitation, analysis, specification, and validation of software requirements. It is widely acknowledged within the software industry that software engineering projects are critically vulnerable when these activities are performed poorly.
2. Software design is the software engineering life cycle activity in which software requirements are analyzed in order to produce a description of the software's internal structure that will serve as the basis for its construction.
3. The term software construction refers to the detailed creation of working, meaningful software through a combination of coding, verification, unit testing, integration testing, and debugging.
4. Software testing consists of the dynamic verification of the behavior of a program on a finite set of test cases, suitably selected from the usually infinite executions domain, against the expected behavior.

What are the knowledge areas?

5. Software maintenance is defined as the totality of activities required to provide cost-effective support to software. Activities are performed during the pre-delivery stage, as well as during the post-delivery stage. Pre-delivery activities include planning for post-delivery operations, for maintainability, and for logistics determination for transition activities. Post-delivery activities include software modification, training, and operating or interfacing to a help desk.
6. Configuration management (CM), then, is the discipline of identifying the configuration of a system at distinct points in time for the purpose of systematically controlling changes to the configuration, and maintaining the integrity and traceability of the configuration throughout the system life cycle.
7. Software Engineering Management can be defined as the application of management activities—planning, coordinating, measuring, monitoring, controlling, and reporting—to ensure that the development and maintenance of software is systematic, disciplined, and quantified

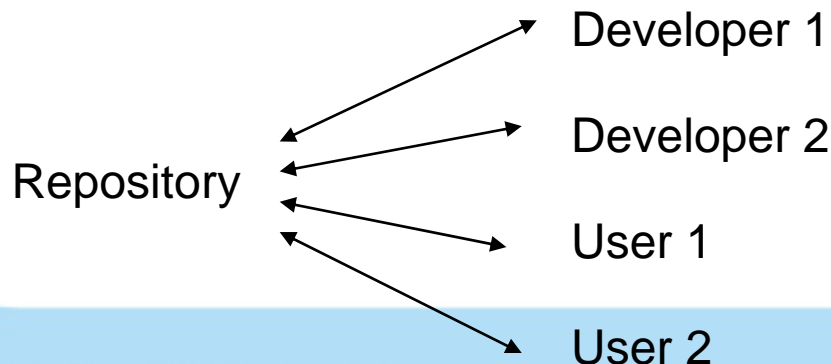
What are the knowledge areas?

8. The Software Engineering Process KA is concerned with the definition, implementation, assessment, measurement, management, change, and improvement of the software life cycle processes themselves.
9. Software development tools are the computer-based tools that are intended to assist the software life cycle processes.
10. Software Quality in this KA of SWEBOK will cover static techniques, those which do not require the execution of the software being evaluated, while dynamic techniques are covered in the Software Testing KA.

Status of our work...

- | | |
|---|-----------------------------------|
| • Software requirements | – ad hoc |
| • Software design | – ad hoc |
| • Software construction | – code reviews |
| • Software testing | – ad hoc |
| • Software maintenance | – ad hoc |
| • Software configuration management | – new, systematic approach |
| • Software engineering management | – in support of SCM |
| • Software engineering process | – ad hoc |
| • Software engineering tools and methods | – subversion for SCM |
| • Software quality | – ad hoc |

Software Configuration Management Tool



- [-] Community_modelling_repository
 - [+] branches
 - [+] tags
 - [+] trunk
- [-] EC Model Development
- [+] GDP
- [+] MEC Development
- [-] mesh_repository
 - [+] branches
 - [+] tags
 - [+] trunk
- [-] model_runs_repository
 - [+] branches
 - [+] tags
 - [+] trunk

Status of our work...

