

EGI: Managing the Software Process

Steven Newhouse, Francesco Giacomini, Oliver
Keeble, Markus Schulz, Aleš Křenek, Achim
Streit, Oxana Smirnova & Farid Ould-Saada
[ARC, gLite and UNICORE]

Remit

- Relationship between EGI.org and the software providers within UMD
 - ARC, gLite, UNICORE
 - Other providers: Globus, dCache, Condor, ...
- Roles and responsibilities of the different actors
 - Relevant staff within EGI.org
 - MCB: Balance between strategy and tactics
- Glossary:
 - MCB: Middleware Coordination Board
 - UMD: Universal Middleware Distribution

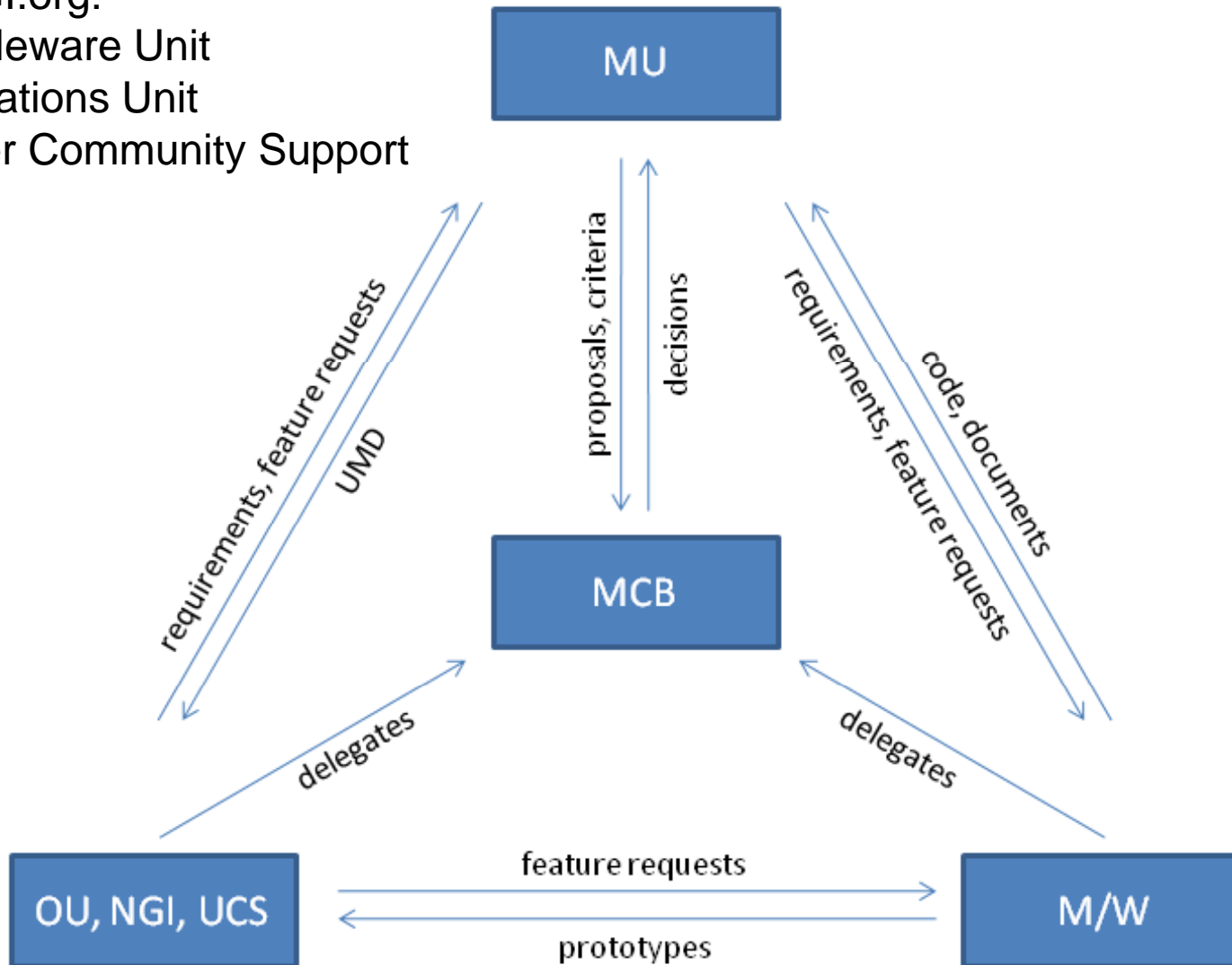
Key Relationships

Within EGI.org:

MU: Middleware Unit

OU: Operations Unit

UCS: User Community Support



Conclusions: Software

- EGI focused on software maintenance and releases
 - Any bugfix/minor issue triaged and assigned to relevant team
 - Response covered by maintenance agreement
 - Any new major/enhanced functionality requests collected and made available to software providers
 - Subsequent development done through non-EGI funding
- Software providers must provide production ready code
 - Able to run and perform in environments defined by EGI
 - Certification burden and responsibility with software provider
 - EGI through MCB defines generic & component-specific criteria
 - Lightweight verification to avoid central release bottleneck

Conclusions: Technical Decision Making

- MCB: Strategic
 - UMD roadmap approval
 - Acceptance criteria
 - Definition of technical policies
- Chief Technical Officer: Tactical
 - Requirement categorisation & prioritisation
 - Day to day technical issues

Next Steps

- Feedback
 - Welcomed now at the meeting
 - Through the wiki page
- Finalising the overview ‘white paper’
 - Discussions at a meeting in Oslo next week
- Drilling down into more details
 - UMD Roadmap: Criteria, scope & initial components
 - Software Life Cycle: Release process
 - Generic and Component Specific Acceptance Criteria