A Review of Standards and Tools for the Engineering Process of Protection Automation and Control Systems

Iony Patriota de Siqueira, D.Sc.
Managing Director
Tecnix Engineering and Architecture Ltd.
Players and Products of an Engineering Cycle

1. Planning
2. Specification
3. Design
4. Deployment

Players:
- Client
- Specifier
- Designer
- Builder
- Tester

Products:
- Requirements
- Specification
- Design
- Deployment
Languages for the Engineering Cycle

1. NL – Natural Language
2. CNL – Controlled Natural Language
3. IEC 61850 – Power Utility Automation
4. IEC 61499 – Function Blocks
5. IEC 61131 – Programmable Controllers
6. IEC 13568 – Information Technology
7. OMG UML – Unified Modeling Language
8. OMG SysML – System Modeling Language
IEC 61850 Engineering Design Cycle

1. User Requirement
2. System Specification Tool
   - System Specification Description
3. System Configuration Tool
   - Substation Configuration Description
   - IED Capability Description
4. IED Configuration Tool
   - Instantiated IED Description

SED
ICD
IID
SSD
SCD
CID
IED
IED
Language Adequacy Criteria for Requirements

- **Applicability** – to power systems, with reasonable assumptions and vocabulary
- **Implementability** – by a reasonable path to code or implementation technology
- **Testability** – by computer / human simulating the requirement with input signals
- **Checkability** – by human and computers for accuracy and completeness
- **Maintainability** – for supporting documentation, conservation and modifications
- **Modularity** – by composition operators for modules, structures, and inheritance
- **Expressibility** – to describe all kinds of needs and requirements of a system
- **Soundness** – for detecting inconsistencies and ambiguities in requirements
- **Verifiability** – to prove that a specification or project attains its requirements
- **Usability** – for directly reading and writing by human experts and computers
- **Tools** – availability in quality, training, support, user base, industrial use
- **Looseness** – allowing incompleteness or non-determinism in the requirements
- **Learning** – new user quickly learns the concepts, techniques, and heuristics
- **Maturity** – in development, training, certification, user base, impetus
- **Modeling** – for data representation, restrictions, relationships & abstractions
- **Discipline** – to force users to write reasonably well-behaved requirements
## CIGRE Language Adequacy Evaluation

<table>
<thead>
<tr>
<th>Adequacy</th>
<th>Natural Language</th>
<th>Controlled Language</th>
<th>IEC 61850</th>
<th>IEC 61499</th>
<th>IEC 61131</th>
<th>UML</th>
<th>SysML</th>
<th>IEC 13568</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applicability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Implementability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Testability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Checkability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maintainability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modularity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expressibility</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Soundness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Verifiability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tools</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Looseness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learning</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maturity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modeling</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discipline</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Worst** – Lowest level of adequacy of the format by this criterion
- **Bad** – Low level of adequacy of the format by this criterion
- **Fair** – Reasonable level of adequacy of the format by this criterion
- **Good** – High level of adequacy of the format by this criterion
- **Best** – Highest level of adequacy of the format by this criterion
Survey Contributions per Country (>100)
## Language Adequacy to Engineering Cycle

<table>
<thead>
<tr>
<th>Adequacy</th>
<th>Requirements</th>
<th>Specification</th>
<th>Design</th>
<th>Implementation</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applicability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Implementability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Testability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Checkability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maintainability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modularity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expressibility</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Soundness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Verifiability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tools</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Looseness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learning</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maturity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modeling</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discipline</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
How to model check IEC 61850?

1. Requirement
   - Basic Application Profile (BAP)
2. Specification
   - System Exchange Description (SED)
3. Design
   - IED Capability Description (ICD)
4. Configuration
   - Instantiated IED Description (IID)

Model Checker:
- FICS
- SSD
- SCD
- CID
- Verdict
Conclusions

Present
- No standard language for requirements
- Natural language used for requirement
- No formal method for design verification
- No standard tool for requirement specification

Future
- Formal language for requirement specification
- Standard tool for design verification
- Integration to IEC 61850 SCL language
Thanks

Iony Patriota de Siqueira
iony@tecnix.com.br
www.tecnix.com.br