MAP Distributed Co-Simulation Protocol

Martin Krammer
DCP MAP Leader
martin.krammer@v2c2.at

American Modelica Association User Conference
October 11, 2018
Outline

- Introduction
- The Distributed Co-Simulation Protocol (DCP)
  - Communication Protocol
  - Architecture Description
  - Operating Modes
  - State Machine
  - Exchange of Input and Output Data
  - Use Case
- Current Status
- Outlook
Motivation

- The Functional Mock-up Interface (FMI, MODELISAR project) standardizes integration of simulation models, tools and solvers.
- But what about distributed setups?

- Until now, this is done manually.
The ACOSAR Project

- The ACOSAR project
  - Advanced Co-Simulation Open System Architecture
  - Duration: 09/2015-08/2018
  - Costs: 8,123k€
  - Effort: 60 PY

- ACOSAR focuses on integration of
  - Real-time and real-time, and
  - Real-time and non-real-time systems

- Primary goal: Negotiate technical specification of communication protocol intended for standardization
Main design aspects

- **Interoperability**
  - Define a communication protocol
  - Goal: Pursue standardization with a recognized standardization body

- **Compatibility**
  - Support a broad range of systems, from small microcontrollers to large test rigs
  - Targets: Low overhead, low memory footprint

- **Integration**
  - Develop methodology for application in development processes
  - Master-Slave concept

- **Communication**
  - Support multiple transport protocols
  - Initially: UDP, CAN, USB, Bluetooth, and EtherCAT

- **Economy**
  - Reduce development time
  - Decrease computing cost
  - Accelerate time-to-market
The Distributed Co-Simulation Protocol

Architecture Description

- Communication protocol
- Standardized mapping
- Non-standardized mapping
- Transport protocol PDU
  Payload: DCP PDU
  - native DCP
  - non-native DCP
- Transport protocol PDU incl. DCP PDU
- Standardized mapping description
The Distributed Co-Simulation Protocol

- Default integration methodology
  - Defines provider-integrator relationship
  - Relies on DCP slave description file (.dcpx)
Operating Modes

- The DCP covers three different time domains

<table>
<thead>
<tr>
<th>Operating mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soft real-time (SRT)</td>
<td>Simulation time should be synchronous to absolute time, tolerant against RT violations.</td>
</tr>
<tr>
<td>Hard real-time (HRT)</td>
<td>Simulation time synchronous to absolute time, intolerant against RT violations.</td>
</tr>
<tr>
<td>Non-real-time (NRT)</td>
<td>Independent from absolute time.</td>
</tr>
</tbody>
</table>
The Distributed Co-Simulation Protocol

- DCP slave state machine for simulation control
- A typical simulation cycle
  1. Registration
  2. Configuration
  3. Initialization
  4. Run/Compute
  5. Stop
  6. (Error)
## Relationship of DCP to other Modelica Association Projects

### DCP and FMI: Commonalities and Differences

<table>
<thead>
<tr>
<th>Aspect</th>
<th>FMI</th>
<th>DCP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Character</td>
<td>Specification for advanced programming interface (API)</td>
<td>Specification for application level communication protocol</td>
</tr>
<tr>
<td>Specification focus</td>
<td>Slave only</td>
<td>Slave only</td>
</tr>
<tr>
<td>Entity</td>
<td>FMU (software component)</td>
<td>DCP slave (hardware and/or software)</td>
</tr>
<tr>
<td>Integration aid</td>
<td>modelDescription.xml</td>
<td>DCP slave description file (.DCPX)</td>
</tr>
<tr>
<td>Controlled by</td>
<td>Master</td>
<td>Master</td>
</tr>
<tr>
<td>Data exchange</td>
<td>Master only</td>
<td>Slave-master-slave and slave-to-slave</td>
</tr>
<tr>
<td>Execution</td>
<td>Function calls</td>
<td>Exchange of PDUs</td>
</tr>
<tr>
<td>Real-time</td>
<td>n/a</td>
<td>SRT/HRT/NRT</td>
</tr>
<tr>
<td>Logic</td>
<td>State machine driven</td>
<td>State machine driven</td>
</tr>
<tr>
<td>Language</td>
<td>C</td>
<td>Language independent</td>
</tr>
<tr>
<td>Reference code</td>
<td>C</td>
<td>C++</td>
</tr>
<tr>
<td>Communication layer</td>
<td>n/a</td>
<td>UDP, CAN bus, USB, Bluetooth</td>
</tr>
</tbody>
</table>
Future of DCP - Standardization

- The DCP 1.0-RC1 was submitted to **Modelica Association** for standardization
- Will be maintained as **Modelica Association Project (MAP)**
- Most recent version is DCP 1.0-RC3
- Sustainable ACOSAR project result
  - The DCP will be freely available
  - Open for everyone!
- Website: [www.dcp-standard.org](http://www.dcp-standard.org) (coming soon)
Future of DCP - Roadmap

**Release**
- Turn RC (release candidate) document into Standard 1.0 document
- Release final DCP 1.0.0 specification document (expected EO Q4/2018)
- Set up steering committee/advisory committee, management of visitors
- Move from ACOSAR to MA infrastructure
- Next design meeting in 3 weeks

**Dissemination**
- website [www.dcp-standard.org](http://www.dcp-standard.org)
- conferences and events, e.g. Modelica 2019 conference [https://www.modelica.org/events/modelica2019](https://www.modelica.org/events/modelica2019)

**Goals**
- Establishment of long-term activities
- Establishment of new working groups, as needed
- Discover new DCP applications
- Management of end-user feedback
- Support additional transport protocols
Any questions?
Martin Krammer
martin.krammer@v2c2.at