Weapon Simulation as a Service for the entire tactical cycle

2024-SIW-Presentation-005

Arno Gerretsen, Royal Netherlands Aerospace Centre (NLR)
Nico Zink, Royal Netherlands Aerospace Centre (NLR)
arno.gerretsen@nlr.nl
Outline

• Needs and benefits
• Requirements
• Technical implementation
• Experiences from first usage
• Future developments
• Interface standardization
Why develop a new weapon server?

• NLR has a fighter aircraft research simulator with an existing weapon server, but this server does not allow us to:
  ▪ Tune the weapon parameters or switch to agreed weapon parameters for an exercise
  ▪ Integrate new weapon models ourselves

• Correlated pre-launch weapon information
  ▪ Is crucial for pilot to tactically deploy a weapon
  ▪ Lack of correlation between pre-launch information and weapon flyout hinders pilot in using proper tactics
  ▪ Weapon server should also provide pre-launch information
Benefits of using a weapon server

• **Modularization**
  ▪ Add or replace weapon model without impacting the entire simulation
  ▪ Easier to integrate same weapon into multiple simulations
  ▪ Dedicated team can develop weapon server

• **Fair-fight in distributed simulation**
  ▪ All participants use the same weapon model
  ▪ Allows consistent tactics within the exercise
Requirements

- **Functional**
  - Simulate weapon flyout in real-time
  - Calculate pre-launch and post-launch information fast enough for simulation client
  - Interact with simulation environment via DIS
  - Able to handle multiple flyouts and information requests in parallel
  - Able to serve multiple simulation clients

- **Extendibility**
  - Minimal development effort to integrate new weapons
  - Support multiple protocols for communication with clients

- **Usability**
  - Configure weapons and switch weapon parameters without development effort
  - User friendly to start and monitor the application
Technical architecture

Client Request / Response

Services

- Flyout simulation
- DLZ calculation
- Timing calculation

Simulation environment

EntityState
- Fire
- Detonation
- ...

Frontend

HTTPS

DIS

DIS

DIS

WeaponServer library

WeaponServer library

WeaponServer library
Calculation services

- **Flyout service**
  - Real-time (50 Hz) flyout of weapon model
  - High precision timer
  - Target input from simulation environment
  - Output to simulation environment
**Calculation services**

- **DLZ service**
  - Multiple weapon flyouts needed to calculate all DLZ parameters
  - DLZ information is returned to requestor once all calculations are done
    - *Design update rate 2 Hz*
  - Only one request per client processed at a time
Calculation services

- **Timings service**
  - Single weapon flyout to calculate all parameters
  - Timing information is returned to requestor once calculation is done
    - Design update rate 2 Hz
  - Multiple weapons in flight possible for each entity
Frontend

• End user can monitor the status of the weapon server
  ▪ System and simulation environment connection status
  ▪ Active flyouts and calculations
  ▪ Log events

• Accessible via web interface
Lessons learned: Usage in simulation exercise

Weapon server was first used as part of a fighter aircraft simulator in an LVC exercise

- **Easy to switch performance parameters of weapons**
  - Weapon parameters were replaced by parameters agreed within exercise

- **Easy to add new weapon model**
  - GPS-guided bomb was added for this exercise

- **Short development timescale and lack of representative test hardware caused performance issues**
Future extensions weapon server

- **Support additional weapon types**
  - Laser seeking weapons
  - Radiation seeking weapons
  - Network enabled weapons

- **Implement A/G pre-launch information algorithm**

- **Support DIS 7**

- **Support multiple protocols for requests/responses**
  - E.g. network package, web API, Universal Armament Interface (UAI)

- **More generic approach to include weapon models**
  - E.g. Functional Mockup Interfaces/Units (FMI/FMU)
Interface standardization

- Interface standardization is needed to be able switch weapon server instance easily
  - Common weapon server within a distributed simulation exercise
  - Weapon service provided by e.g. weapon manufacturer

- Standardize data model for requests and responses
  - Leave communication protocol an implementation choice
  - Replace sensitive parameter by generic parameters

- Other MSaaS services have similar standardization needs
  - Single standardization activity for common MSaaS services?
## Interface standardization

- **Weapon launch request**
  - A/A missile flyout request
    - Shooter position
    - Shooter attitude
    - Target data
    - Pilot selectable missile settings
  - Flyout confirmation
    - Weapon ID
  - Target position update
    - Weapon ID
    - Target position
- **Pre/post-launch information request**
- **Pre/post-launch information response**
- **Weapon launch confirmation**
- **Weapon updates**

<table>
<thead>
<tr>
<th>Pre-launch DLZ request</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Shooter state</td>
</tr>
<tr>
<td>- Relative target state</td>
</tr>
<tr>
<td>- Weapon type</td>
</tr>
<tr>
<td>- Pilot selectable</td>
</tr>
<tr>
<td>missile settings</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LAR response</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Attach zone ranges</td>
</tr>
<tr>
<td>- Attack zone polygon</td>
</tr>
<tr>
<td>- Optimal steering information</td>
</tr>
<tr>
<td>- Weapon timings</td>
</tr>
</tbody>
</table>
Conclusions

• **Weapon simulation as a service has many benefits**
  - Easier integration of new weapon types
  - Easier switching weapon parameters of model
  - Improved fair-fight in a distributed simulation exercise

• **Including pre/post-launch information in the weapon server greatly improves the pilots ability to use weapons in a tactical manner**

• **Standardization of weapon server interfaces would facilitate rapid switching between weapon server instances**
  - Focus on data model for requests and responses