[dev@insupport.se](mailto:dev@insupport.se)

This solution is using the Milestone SKD API which can be downloaded from the Milestone partner portal.

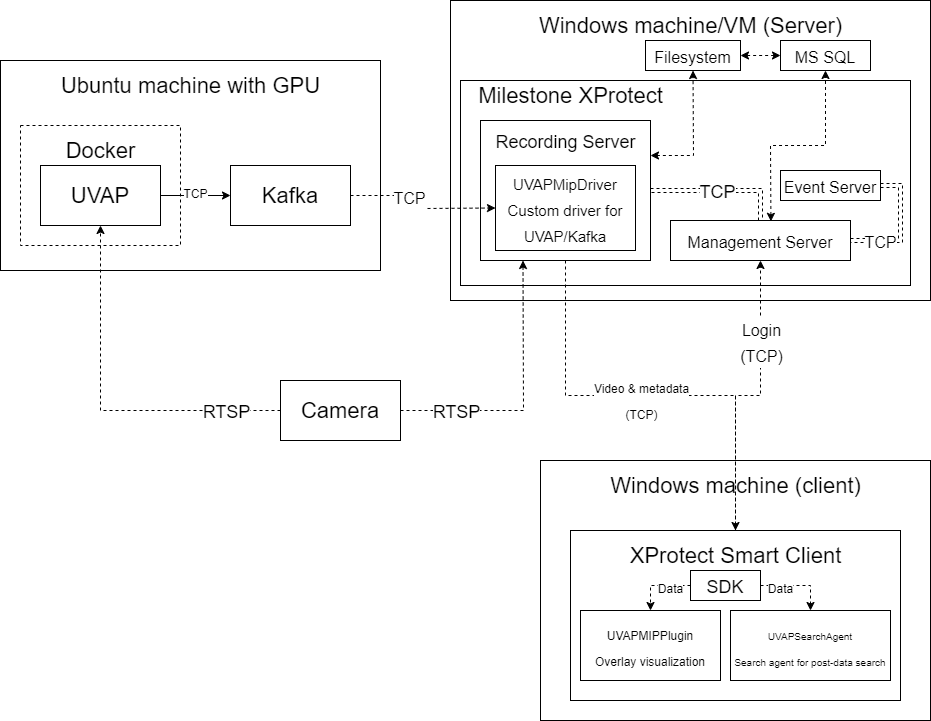
SDK Documentation: <https://doc.developer.milestonesys.com/html/index.html>

Most of the code is based on the samples Milestone provide via the SDK. Here’s some useful samples: Demo Driver, Search agents, Analytics Overlay

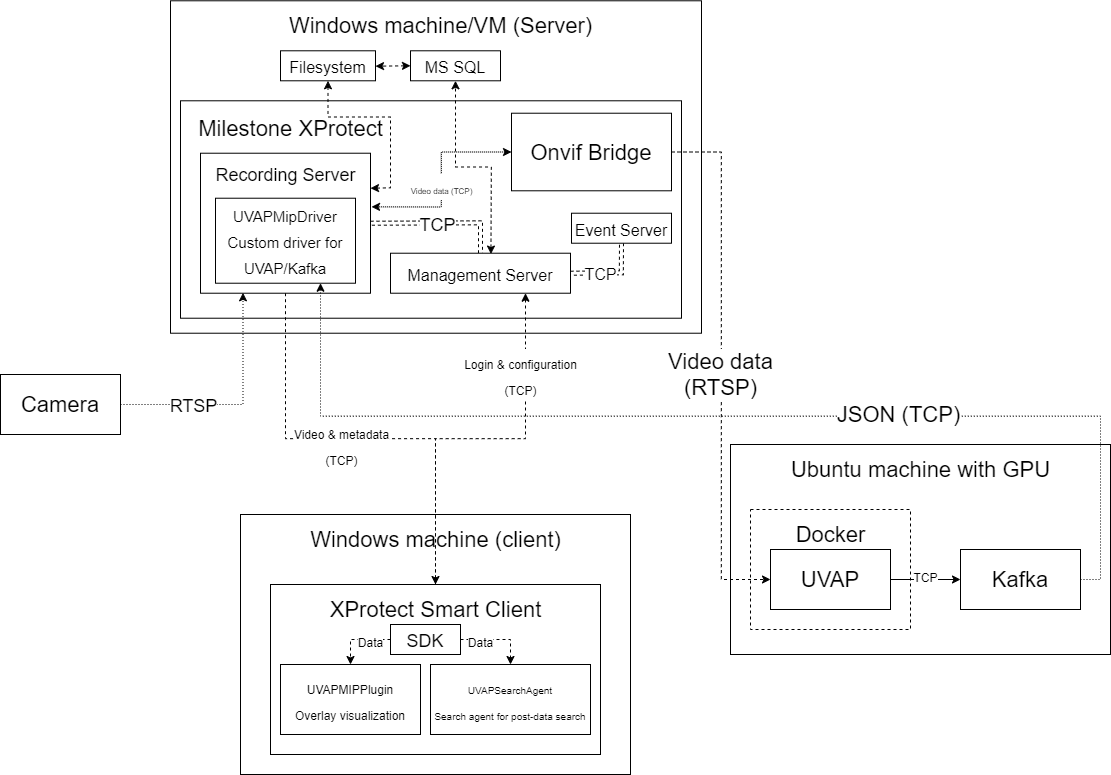
## System interaction

Example 1, camera streams to the UVAP and Milestone

The camera outputs two streams, one to the UVAP and one to Milestone. This doesn’t require the Onvif Bridge and works out of the box.



Example 2, camera streams to Milestone and then to the UVAP via the Milestone Onvif Bridge. This saves bandwidth between the camera and servers but requires the Milestone Onvif Bridge to be installed and configured. This is the only alternative if the camera only supports to output one rtsp stream.



# Driver

The driver implementation is based on the MIPDriver Visual Studio template.

The driver configuration is set by UVAPMipDriverConfigurationManager.cs.

This configuration can **only be fetched from within** the driver. See UVAPMipDriverStreamManager.cs and CreateSession where the configuration is fetched.

Due to this, there’s no way to verify if the metadata devices are from the UVAPMipDriver from other systems/plugins. Therefore, the integrator must rename the metadata devices to ‘head’ or ‘skeleton’ so the UVAPMipPlugin overlay knows what devices to try to visualize. See the method Instance\_NewImageViewerControlEvent in VisualizerBackgroundPlugin.cs. See the driver document for more information on how to do this.

With that said, it’s possible to create the visualizers automatically and have them close if they fail to parse the metadata but I went with this option for now.