



میان افزار در شبکه های حسگر بیسیم

MIDDLEWARE IN WSN

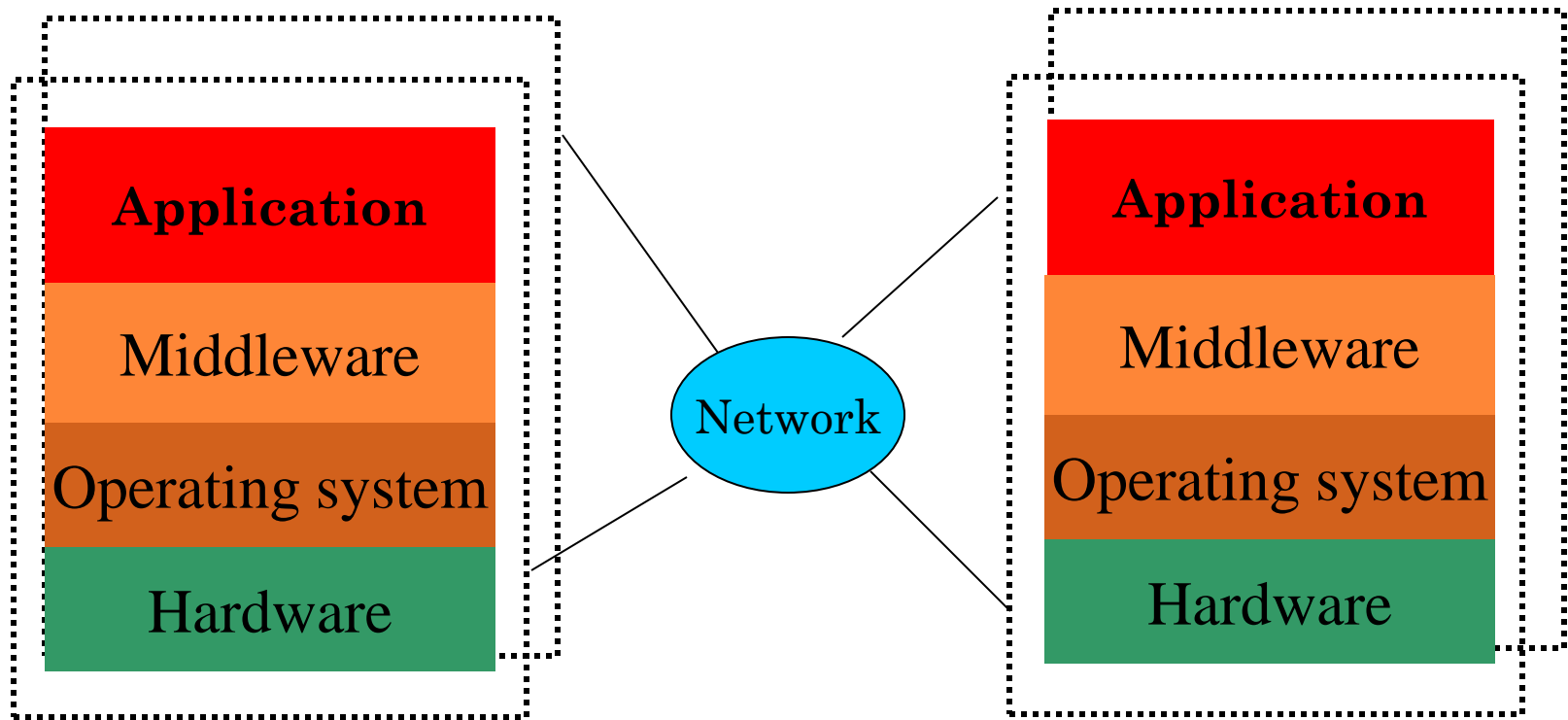
محمد حسین یغمایی مقدم
دانشگاه فردوسی مشهد

تعاریف میان افزار

1. نرم افزاری است که بین برنامه کاربردی و شبکه واقع می‌شود و ارتباط بین کاربردهای مختلف را از طریق بسته‌های متفاوت محاسباتی، مدیریت می‌کند.
2. نرم افزاری است که به عنوان یک واسط بین کاربردها عمل می‌کند بطوری-که بتوانند داده را تبادل کنند.
3. نرم افزاری است که به کاربرد اجازه می‌دهد تا با دیگر نرم افزارها بدون نیاز به اینکه کاربر از کدام لایه‌های زیرین اطلاع داشته باشد، عمل کند.
4. نرم افزاری است که برای مجتمع کردن سیستم‌های سخت-افزاری و/یا نرم-افزاری جدا طراحی شده است. همچنین ارتباط بین سیستم‌های جدا را فراهم می‌کند.
5. لایه ارتباطات است که به کاربردها اجازه می‌دهد تا با محیط‌های شبکه از طریق سخت افزار ارتباط داشته باشند.



COMMON ARCHITECTURE



خصوصیات و چالش‌های میان افزار

○ مدیریت منابع و توان محدود

- میان-افزار باید مکانیزم-هایی را تدارک ببیند تا در خلال مخابره توان پائین، از توانایی پردازش و حافظه، بصورت کارا بهره-مند شود.

○ مقیاس-پذیری، حرکت و ساختار متغیر شبکه

- اگر کاربردی توسعه پیدا کند، میان-افزار باید قابلیت انعطاف لازم جهت اجازه به این توسعه را در هر لحظه و هر جا، بدون تأثیرگذاری بر کارایی شبکه را بدهد.
- همچنین باید در مقابل نقص، حرکت موانع و تداخل مقاوم بوده و مکانیزم-های تحمل خطا، خودپیکربندی و خودنگهدارندگی را دارا باشد.



خصوصیات و چالش‌های میان افزار

○ ناهمگونی

- میان افزار باید مکانیزم-هایی را دارا باشد تا بتواند با انواع متفاوت سخت-افزار و شبکه ارتباط پیدا کند.

○ سازمان پویای شبکه

- از آنجائیکه شبکه-های حسگری، منابع پویایی نظیر: انرژی، پهنای باند و توان پردازشی دارند در طراحی پروتکل‌ها، دانستن منابع شبکه که بصورت دینامیکی تغییر می-کنند ضروری بنظر می-رسد.



خصوصیات و چالش‌های میان افزار

○ فراگیری

- اکثر کاربردهای شبکه-های حسگری، زمان حقیقی می-باشند که در آنها پارامترهای زمان و فضا از اهمیت ویژه ای برخوردار هستند
- میان-افزار باید خدمات زمان حقیقی را برای پاسخگویی به این نوع کاربردها تدارک ببیند.

○ دانش کاربرد

- در اکثر زمینه-ها، توسعه-دهندگان می-خواهند نیازمندیهای ارتباطی کاربردها را به پارامتری شبکه نگاشت دهند
- این کار باعث تنظیم مونیتورینگ شبکه بصورت دقیق می-شود.
- اکثر میان-افزارهای موجود برای کاربردهایی خاص طراحی شده-اند.
- هرچند که تمایل برانست که محدوده وسیعی از کاربردها را پوشش دهند
- توسعه-دهندگان باید بین جامعیت و وابستگی به کاربرد میان-افزار مصالحه برقرار کنند.



خصوصیات و چالش‌های میان افزار

○ ترکیب داده

- در اکثر کاربردهای شبکه حسگری، افزونگی اطلاعات در گره-ها وجود دارد.
- برای برطرف کردن این افزونگی و در نتیجه کاهش تعداد ارسالها به مقصد، ترکیب داده در داخل شبکه باید جزو ملاحظات اساسی در طراحی میان-افزار باشد.
- با اعمال این ویژگی، صرفه-جویی قابل ملاحظه-ای در منابع شبکه صورت می-گیرد.

○ کیفیت سرویس

- میان-افزار باید مکانیزم-هایی جدید برای حفظ کیفیت سرویس در حین کارکرد ارائه دهد
- حتی هنگامی که کیفیت سرویس درخواستی و وضعیت کاربرد تغییر کند، قابلیت تنظیم خود را داشته باشد.
- همچنین بین معیارهایی کارایی نظیر ظرفیت شبکه یا گذردهی، تأخیر ارسال داده و مصرف انرژی مصالحه برقرار کند.



خصوصیات و چالش‌های میان افزار

○ استفاده آسان

- میزان رهایی کاربر از پیچیدگیهای سطح پایین شبکه، ویژگی استفاده آسان میان-افزار را توصیف میکند
- کاربردهای شبکه حسگری با مقیاس بزرگ، منابع متفاوتی را در بر گرفته که پیچیدگی کد و پیاده سازی برنامه نویسی را برای مدیریت این منابع افزایش می-دهد

○ بازبودن

- طراحی یک میان-افزار باز از رکود و ایستایی جلوگیری می-کند
- مدیریت منابع در میان-افزار باید با میزان در دسترس بودن منابع و تغییرات دیگر وفق یابد.
- استانداردها برای سیستم-های باز ضروری بوده و باید دائماً بروز شوند.



خصوصیات و چالش‌های میان افزار

○ پشتیبانی ویژه

- پیچیدگی هر گره را بصورت مجزا پنهان کرده و یک دید کلی را از شبکه فراهم آورد.
- این میان-افزار می-تواند داده محور، انتشار-تقاضا و بر اساس رویداد و فراهم کننده محدوده وسیع از کاربردها باشد.

○ کارایی

- باید از حیث مصرف انرژی کارا و منبع دوست باشد و قابلیت-های بین لایه-ای برای بهینه-سازی را نیز دارا باشد.

○ قابلیت برنامه ریزی

- فراهم کننده تضمین برای پیکربندی و پیکربندی مجدد باشد و همچنین تولید و توزیع سیاست را در مواقع لازم انجام دهد.



خصوصیات و چالش‌های میان افزار

○ قابلیت تطبیق

- شامل مکانیزم-هایی با خصوصیات وفق-پذیری بوده و البته تطبیق مجدد نیازمند مونیترینگ است.

○ توپولوژی

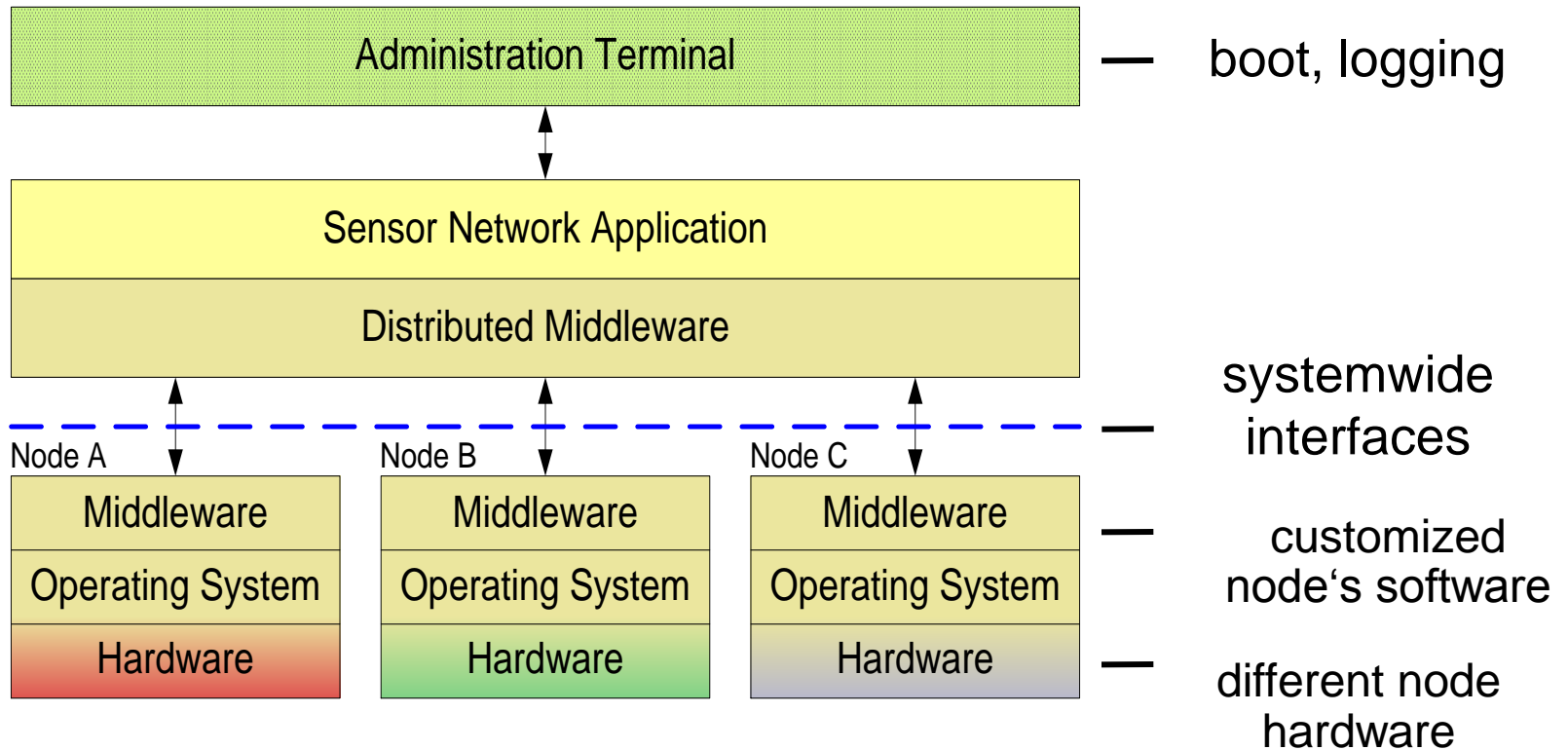
- نوع پیکربندی شبکه، هموار، سلسله-مراتبی و مختلط را مشخص می-کند.

○ امنیت

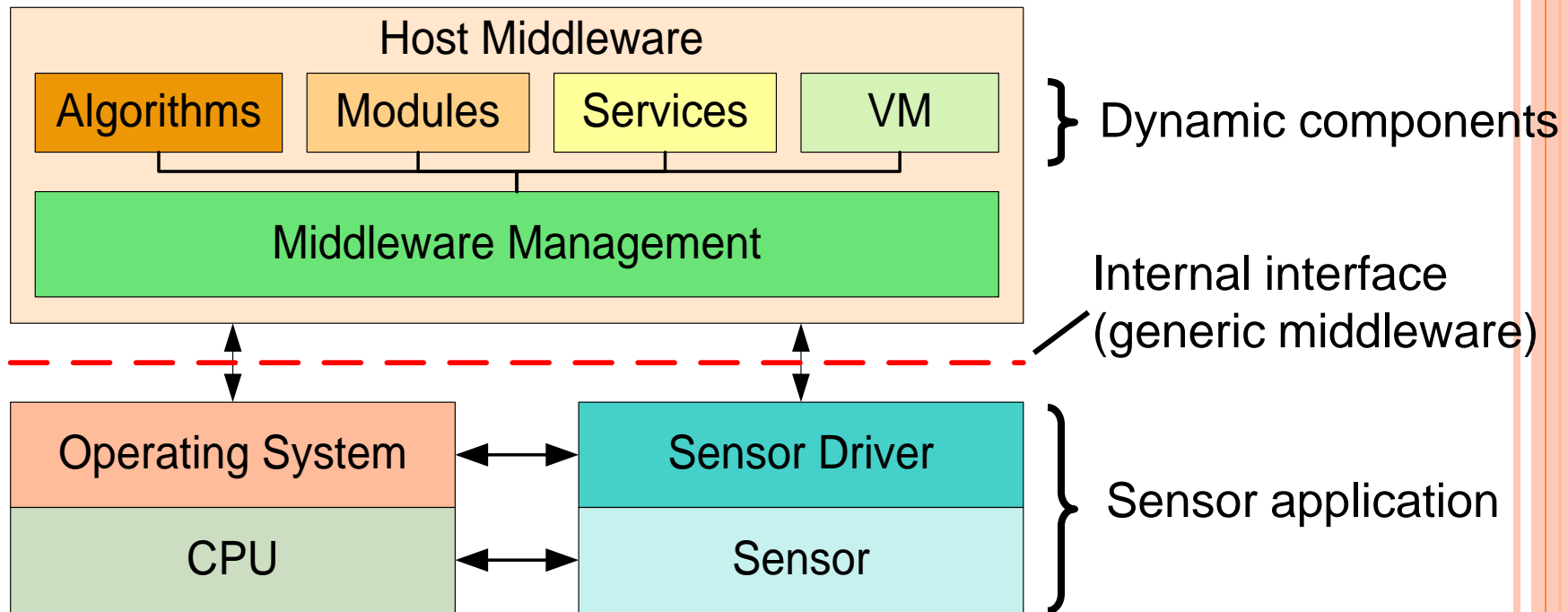
- نسبت به پردازش و مخابره داده-ها و همچنین صدمه دیدن تجهیزات توجه داشته باشد.



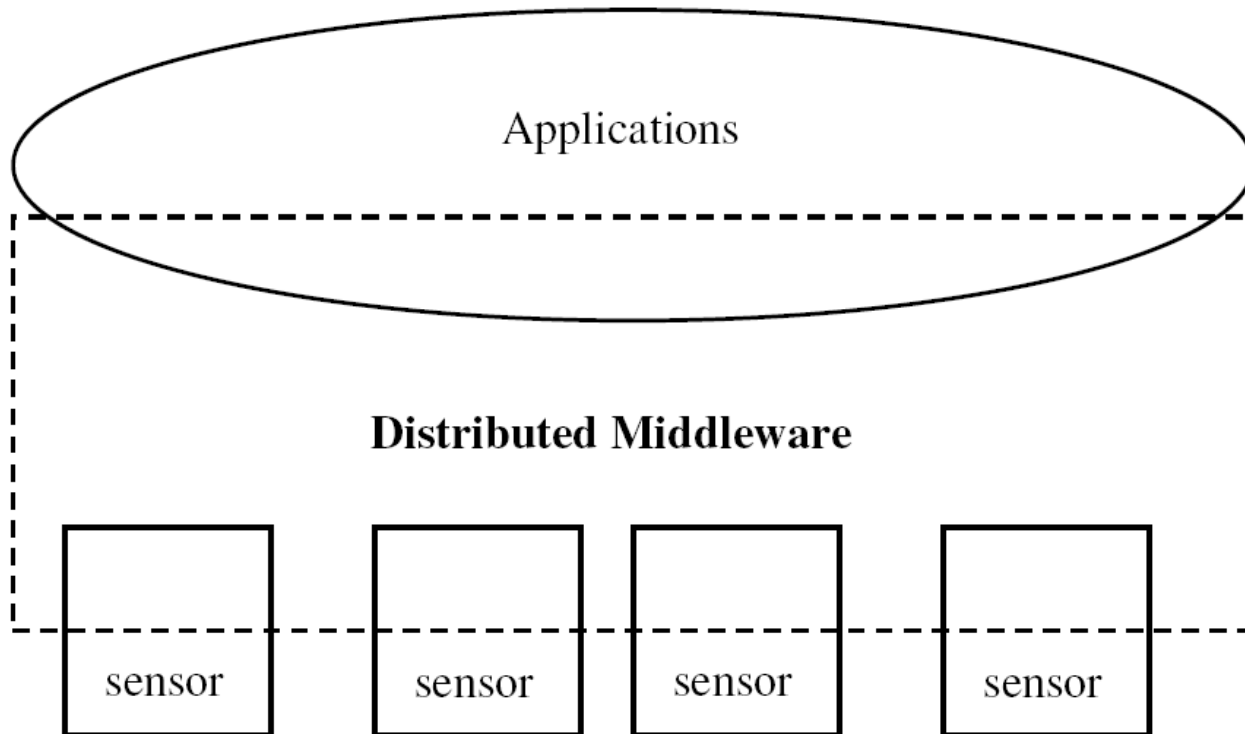
ARCHITECTURE



ARCHITECTURE



ARCHITECTURE



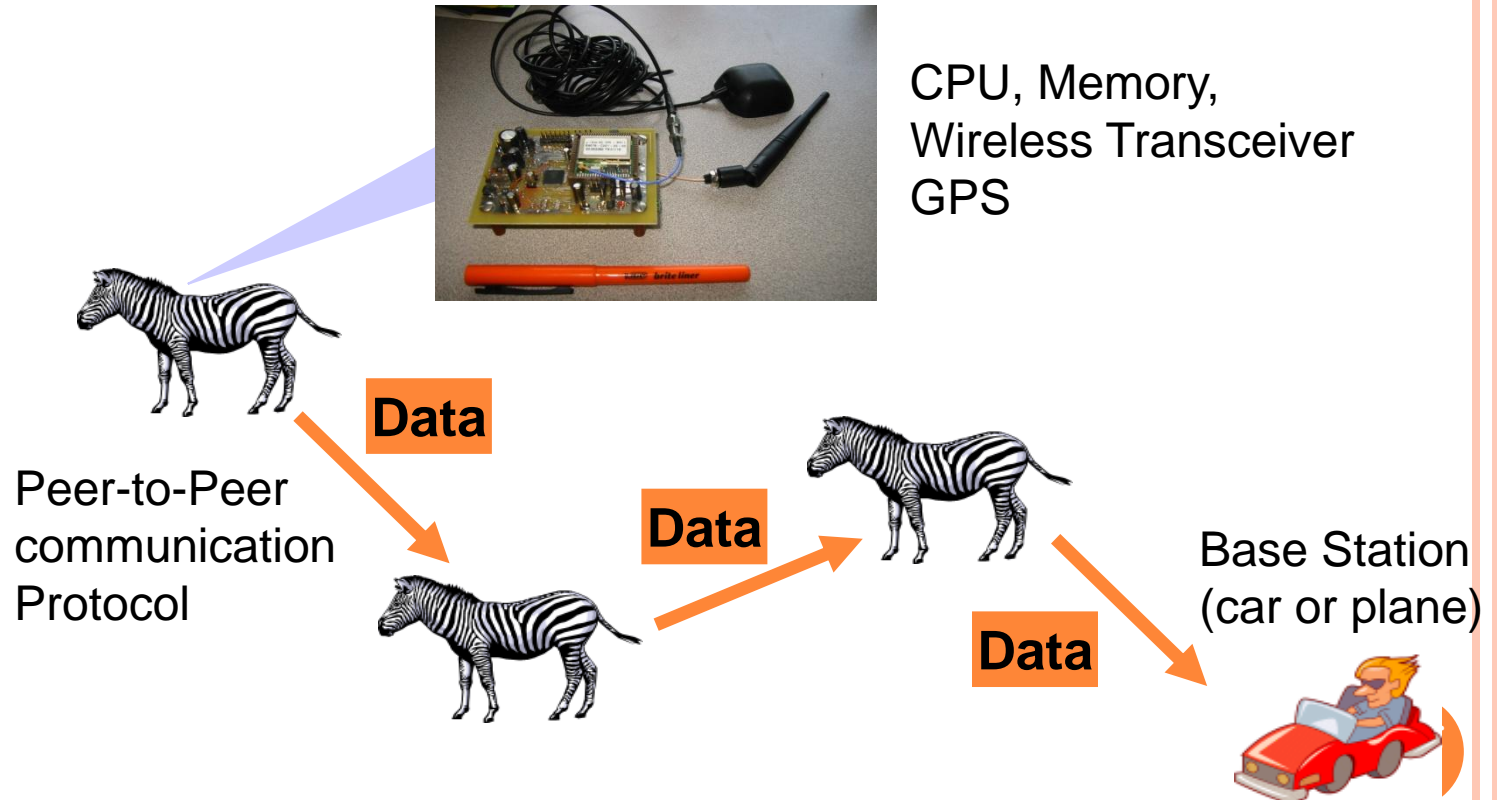
APPROACHES ...

- Application Driven
 - tune network on the basis of Requirement
 - MiLAN
- Message oriented
 - use publish-subscribe mechanism
 - Mire
- Event oriented
 - Request-reply, Garnet
- Service oriented
 - Reflective, Flexible, Service-centric, Service oriented



IMPALA

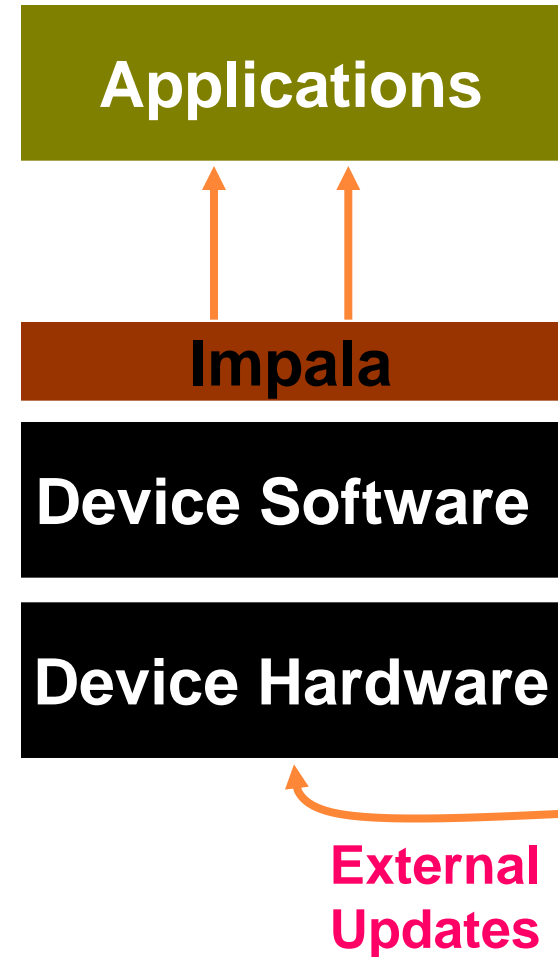
- Zebranet project (long term migration study of wildlife)
- event based programming model



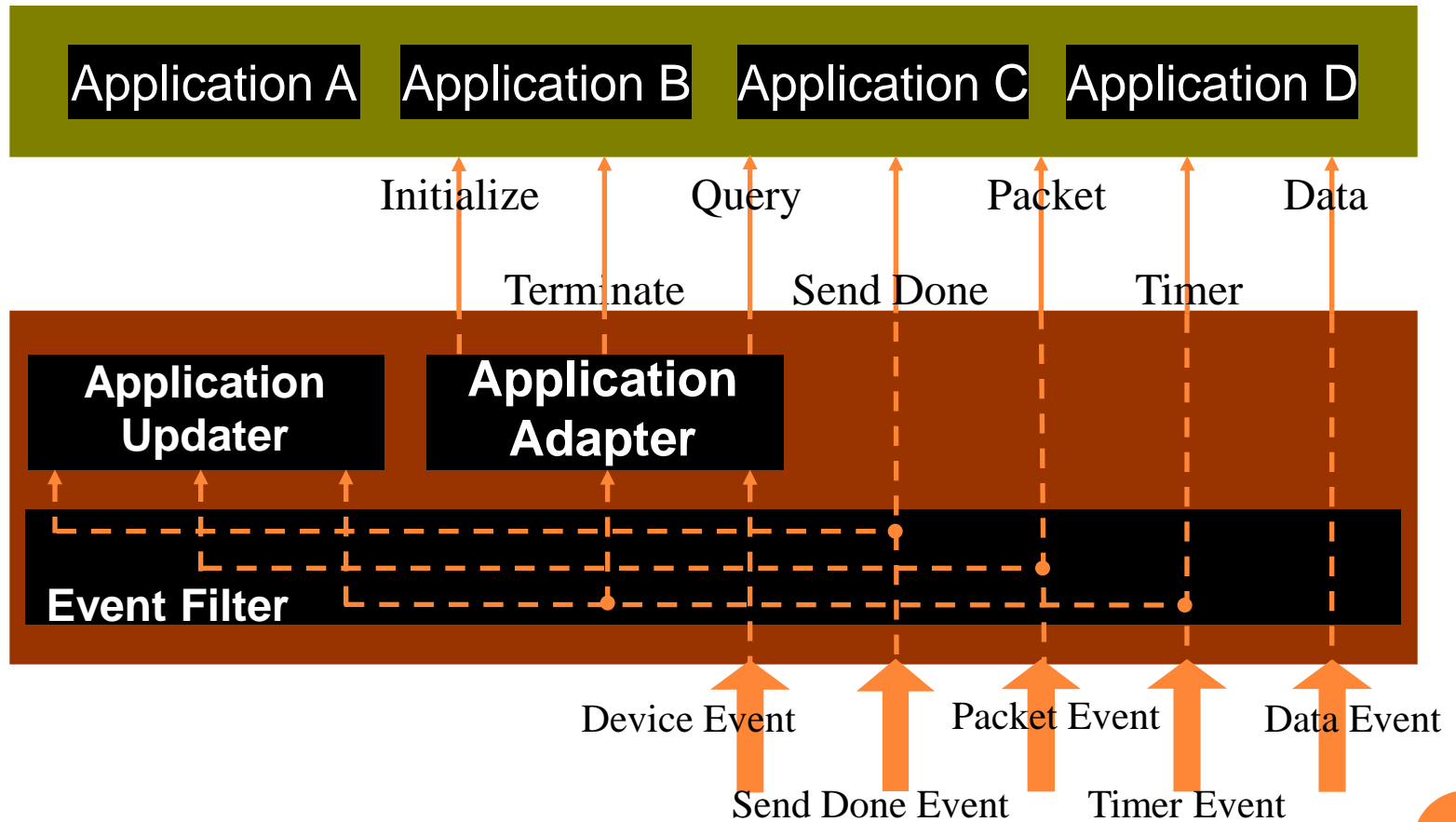
IMPALA FEATURES

- Code modularity
- application adaptability and update
- Fault tolerance
- energy efficiency
- long deployment time

- Not data fusion
- not heterogeneity

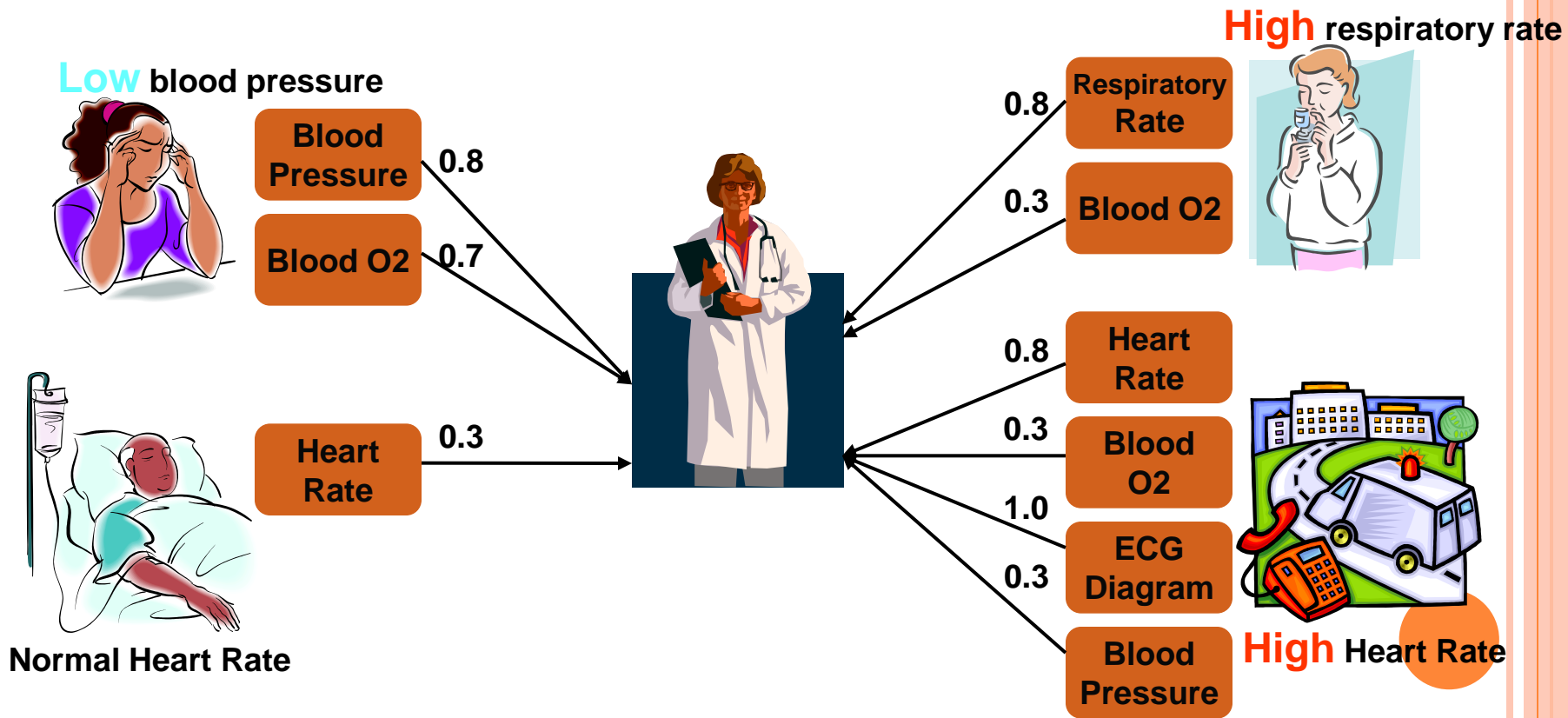


IMPALA ARCHITECTURE



MiLAN

- MiLAN is designed for personal health monitor application

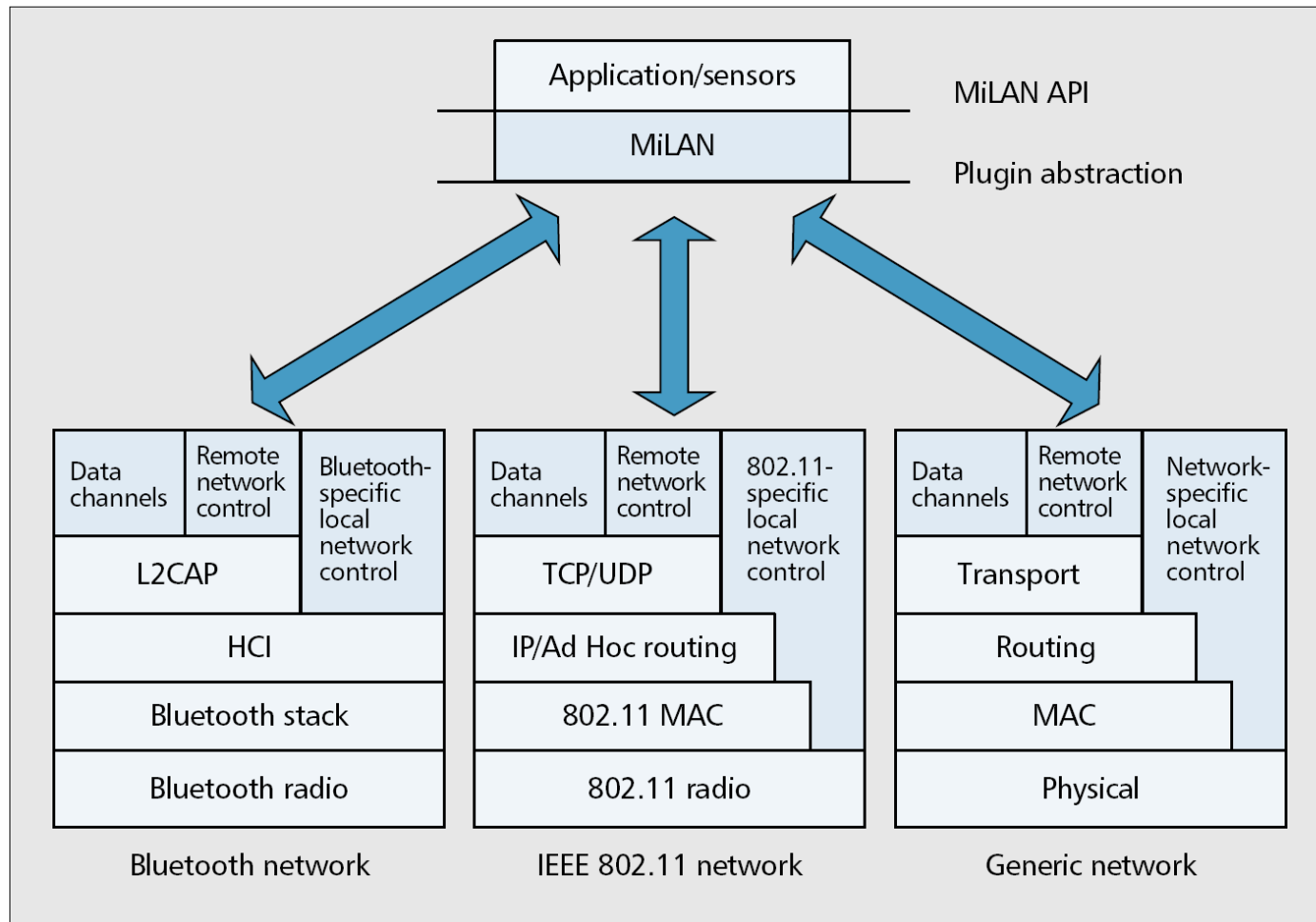


MiLAN

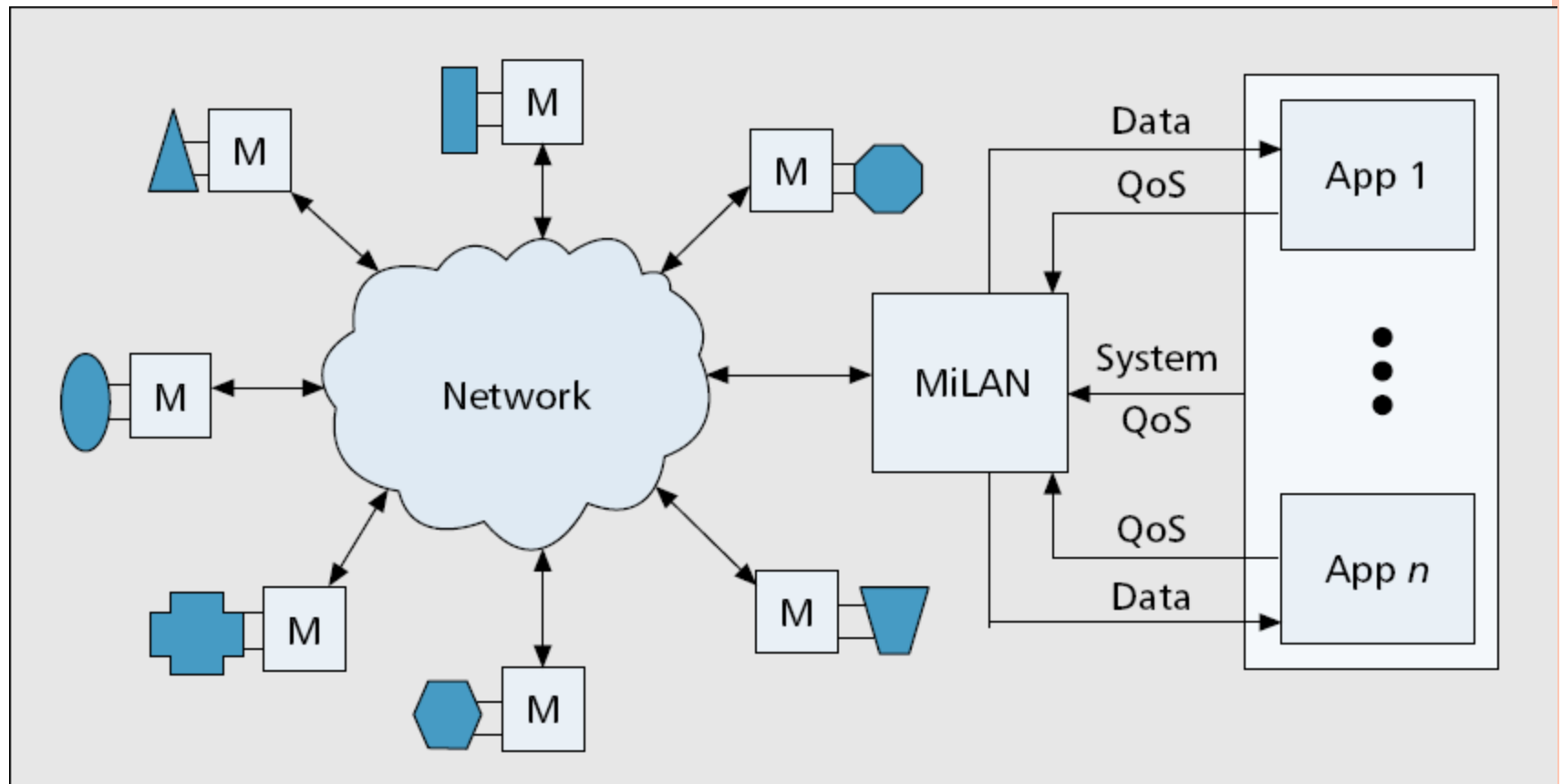
- goals: - maximizing application lifetime - providing application QoS
- MiLAN receives a description of application requirements through specialized graphs
- MiLAN sits on top of multiple physical networks
- convert commands to protocol-specific commands that are passed through the usual network protocol stack



MiLAN ARCHITECTURE



MiLAN ARCHITECTURE



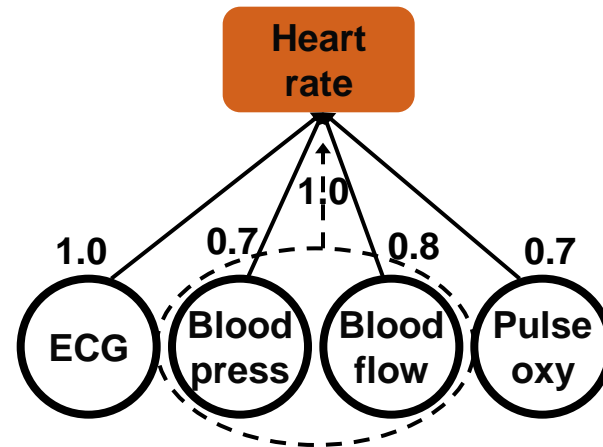
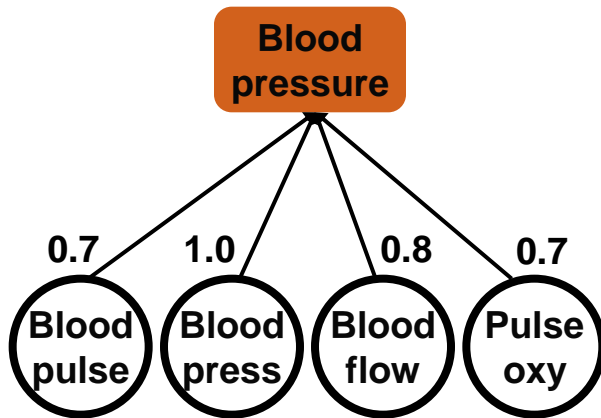
QoS SUPPORT

- The QoS of the different variables depends on which sensors provide data to the application
- application performance can be described by the QoS of different variables of interest to the application
- variables such as blood pressure, respiratory rate, and heart rate may be determined based on measurements obtained from any of several sensors



○ MiLAN must know:

- The variables of interest to the application
- The required QoS for each variable
- The level of QoS that data from each sensor or set of sensors can provide for each variable



CONCLUSION

- There is not a method and tool to analysis and evaluation of Middleware in WSN
- current middlewares can not be used for Real time applications
- most of QoS Features can be apply to middleware specially dependability
 - Reliability
 - Achievability
 - **Safety**

