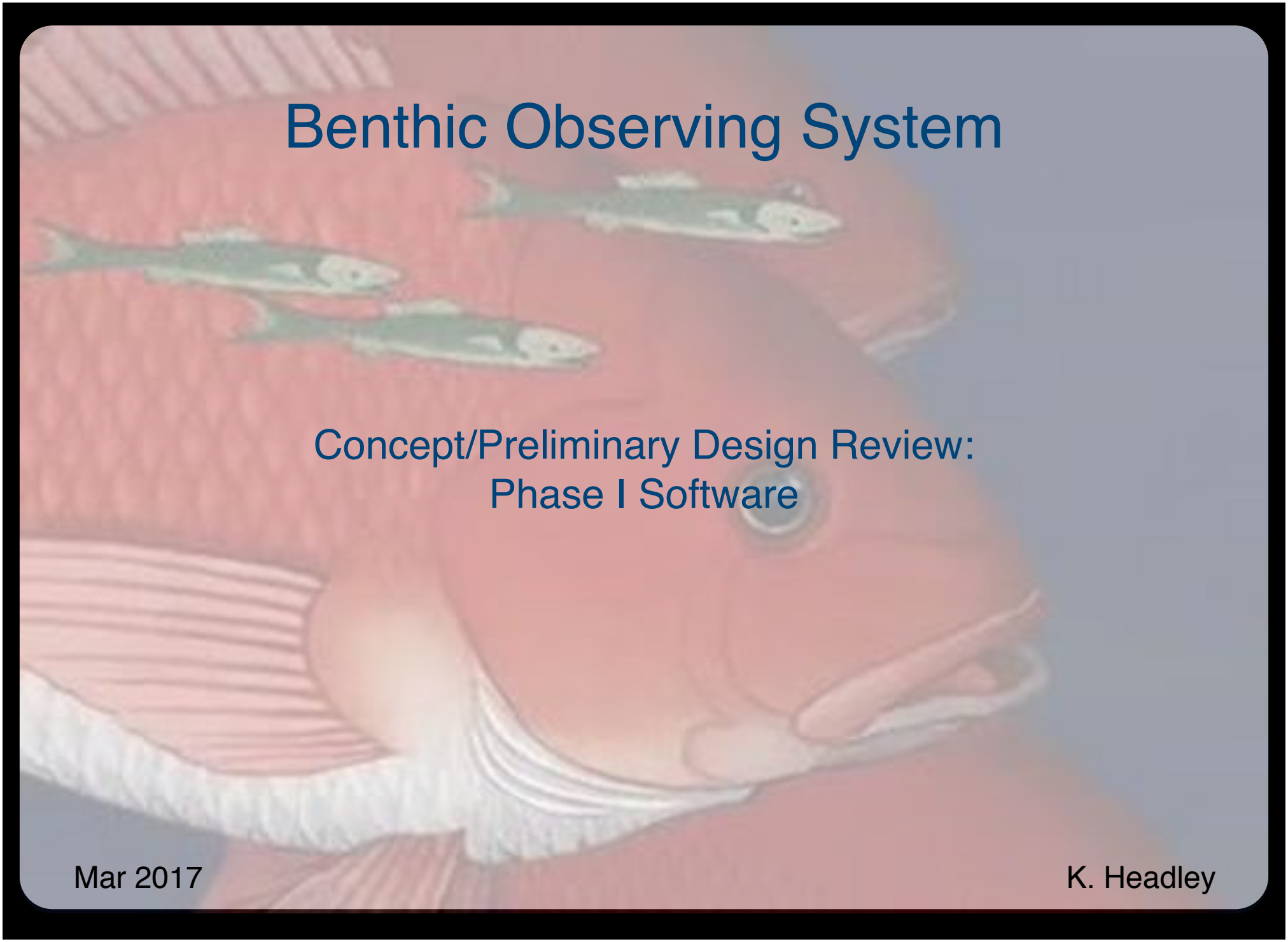


Benthic Observing System

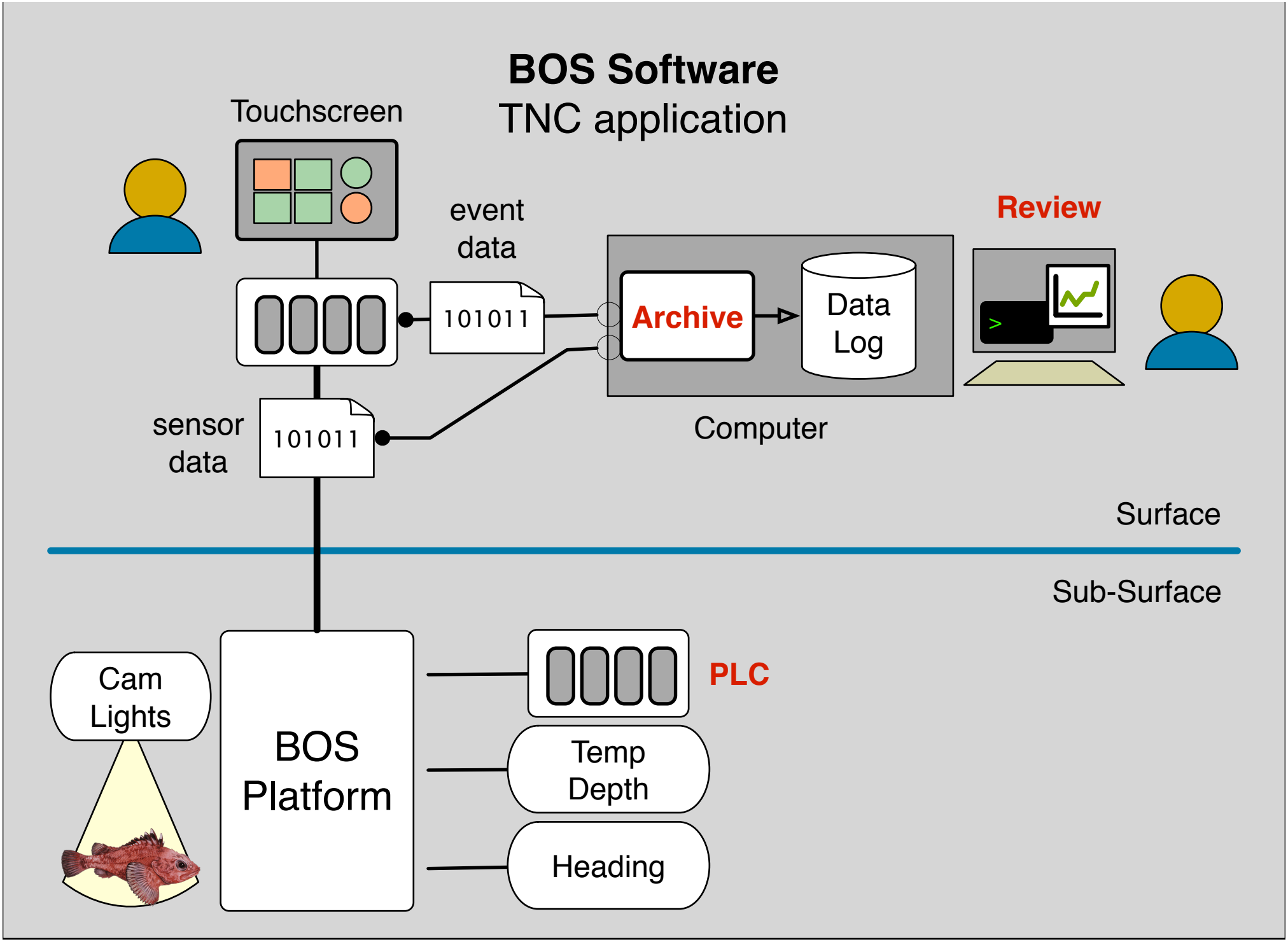
Concept/Preliminary Design Review:
Phase I Software

Mar 2017

K. Headley

Content Outline

- Software in BOS
- Requirements and Features
- Design approaches
- Implementation options
- Selection and recommendations


BOS Software
TNC application
Touchscreen
event
data
Review
101011
Archive
Data
Log
sensor
data
101011
Computer
Surface
Sub-Surface
Cam
Lights
BOS
Platform
PLC
Temp
Depth
Heading

Core Software Requirements

- Archived data streams
  - Platform temperature, depth
  - Platform heading, [pitch, roll?]
- Control events:
  - Camera/lights power
  - start/stop recording)

- Data time-stamps
- Data review (post-op)
- Data monitor (real-time)?
- Ease of use for science users
- Ease of development/extension

Required Software Features

- **Archive**
  - record timestamped data
- **Review**
  - data workflows, post-op

Possible Additional Features

- **Control**
  - system operation
- **Configuration**
- **Telemetry**
  - real-time distribution
- **Interpretation**
  - real-time parsing

Be Careful What You Wish For

Some features introduce requirements that impact approach, schedule and resources:

- Autonomous operation (e.g., battery powered)

- Remote/real time monitoring/control (web, mobile devices)

- GUI

Implementation options need to be evaluated in terms of ***costs and benefits****:*
Schedule, Resources, Risks, Value

Design Goals

- Minimize software development for first iteration
- Architect for future expansion
  - More contextual sensors
  - Real time, distributed/mobile displays
  - Control interface
  - Configuration
  - Autonomous operation
- Keep it Simple: minimal codebase and development environment
- Minimize complexity in the system

Implementation Approaches

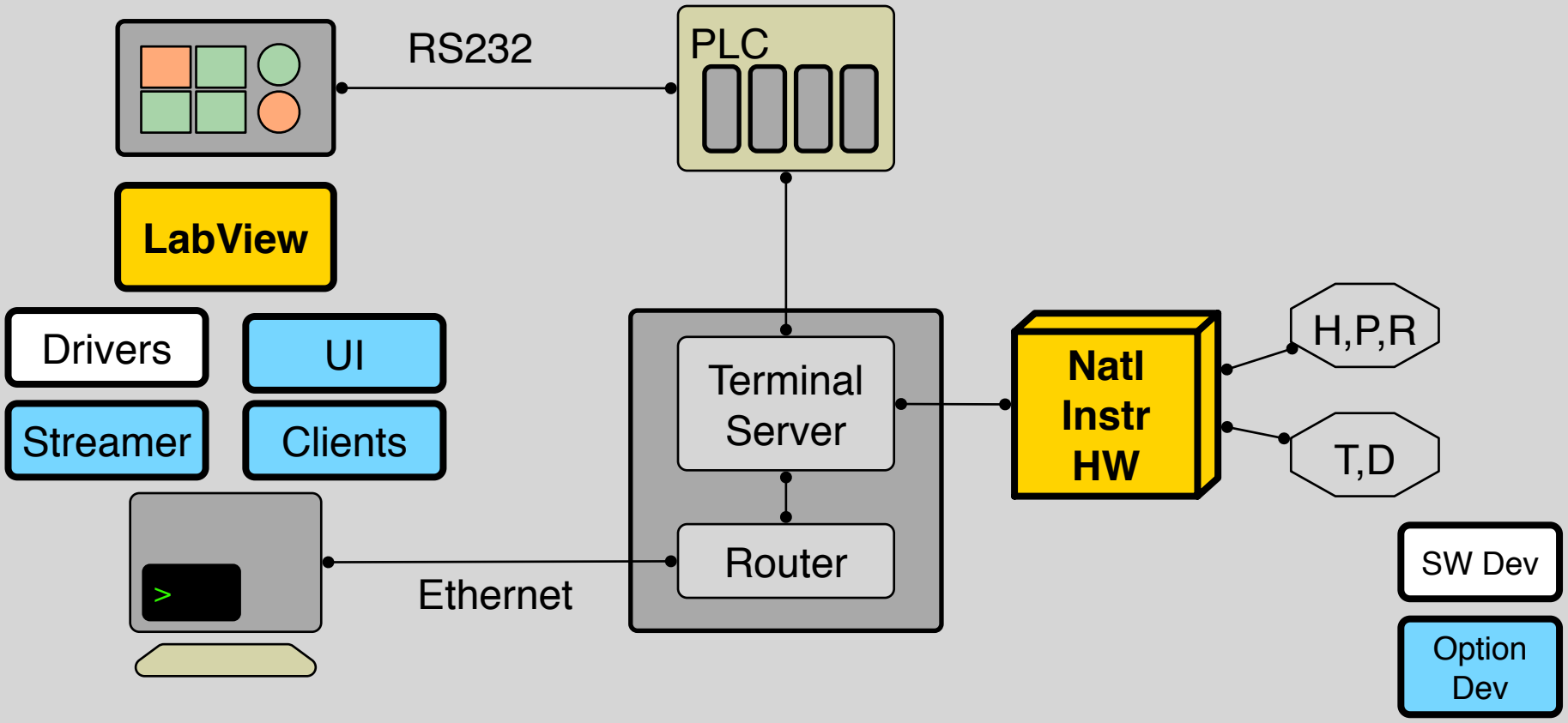
Approaches	COTS	Custom
DAQ: Data acquisition system SW + specialized HW	**Labview/NI** Acumen Campbell Scientific	**OASIS**
Software Solutions SW + PC HW	Sensor mfg utils Comcap **Aggsoft** Magsys	**Flux** (FOCE) SIAM
Low/No Tech a bit of a hack		Internal logging Terminal capture
Custom Solution SW [+ HW?...]	?	BOS app + ?

Use COTS HW and SW to speed development

Concept: LabView

Approach: DAQ

\+ Rapid development
\+ GUI development tools
/ Software development time
\- HW/SW cost
\- Poor fit for autonomous use
? RT performance
? Integration w/ other solutions

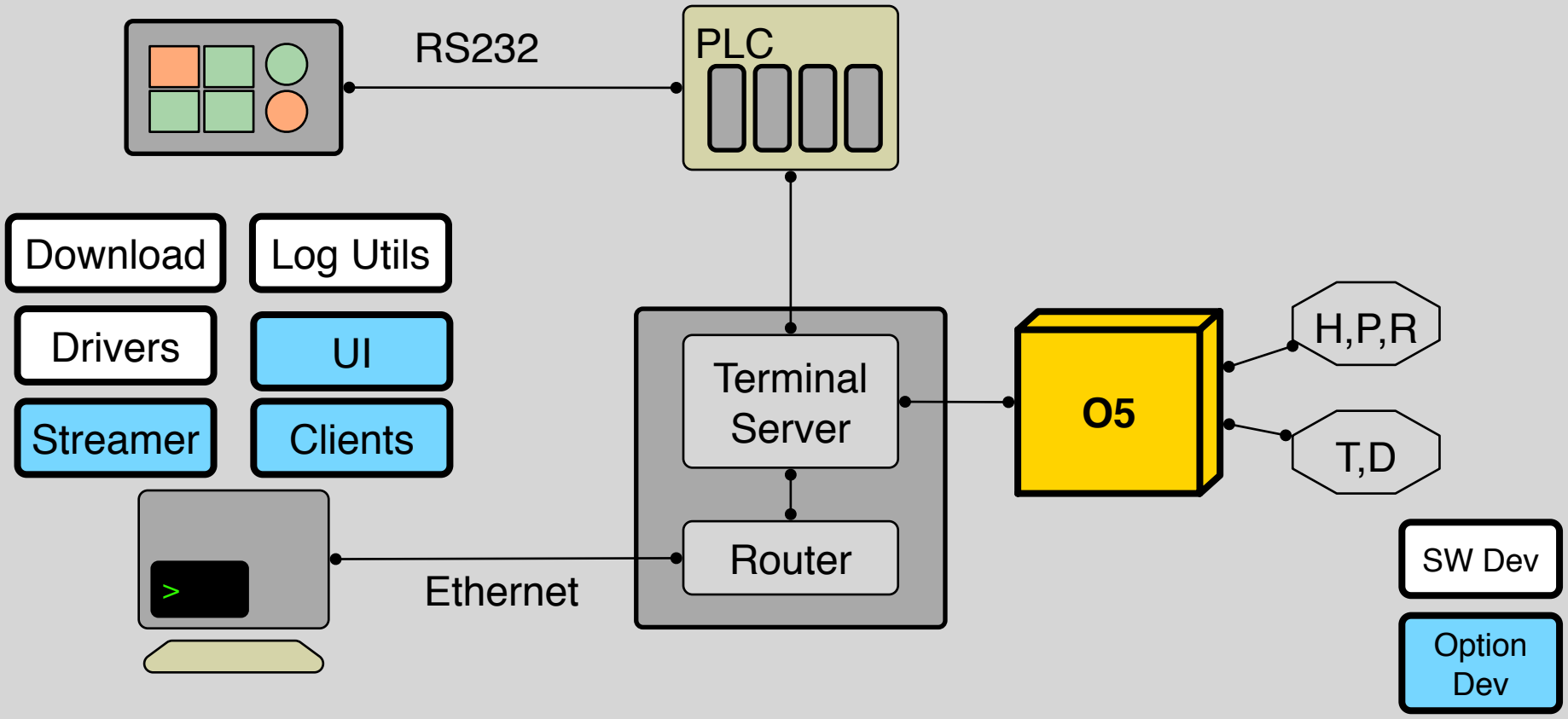


Use existing custom HW and SW with path to autonomy

Concept: OASIS

Approach: DAQ

+ Robust existing HW, SW
+ Autonomous-ready
- Extra integration effort
- RT data distribution, parsing
? max sample rate

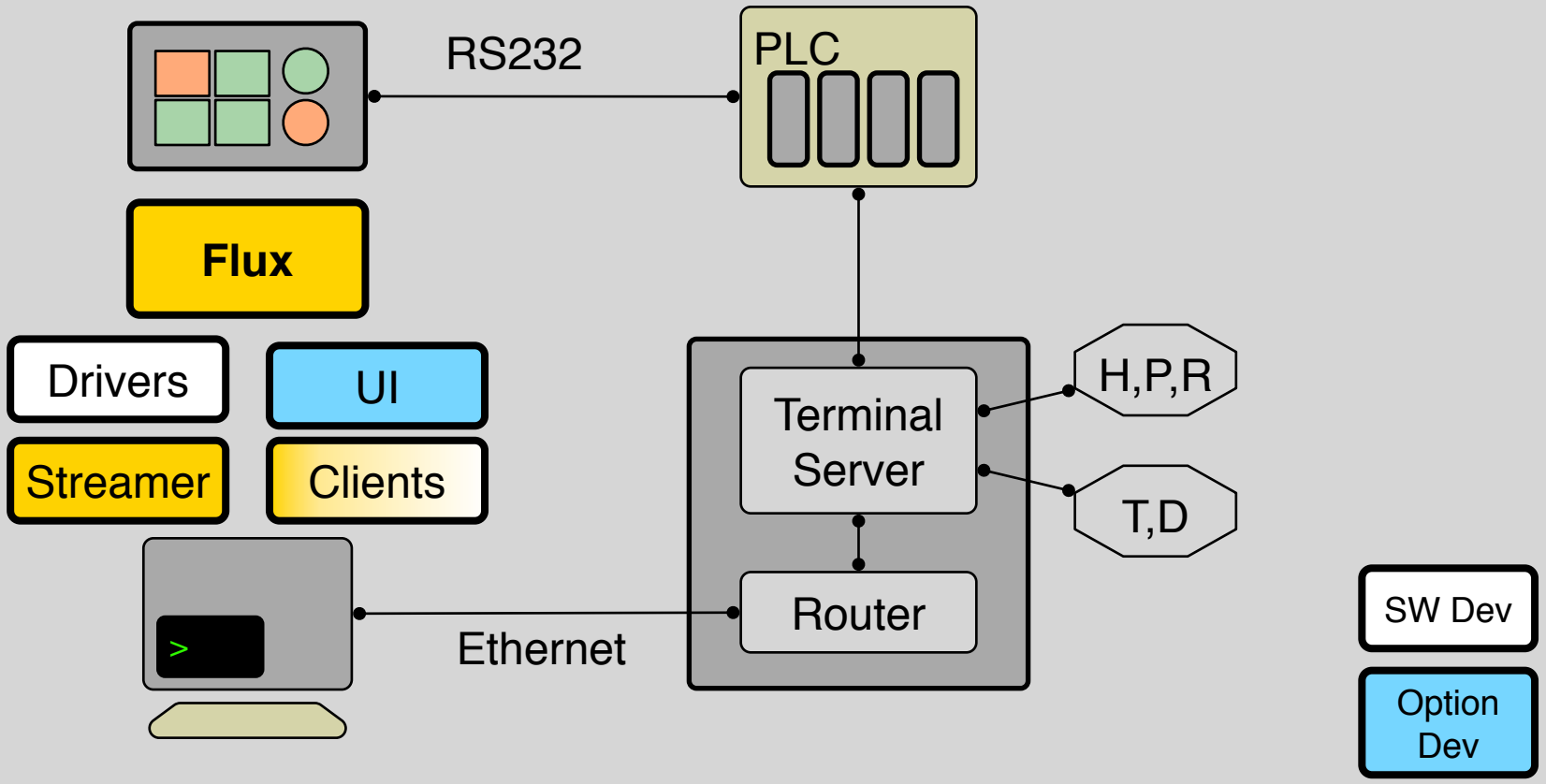


Use existing custom SW-only solution

Concept: Flux

Approach: Software Solution

+ Use commodity hardware
+ Distributed/stand-alone
+ Data streaming, parsing
- Moderate SW development
/ Not autonomous-ready
/ Linux (PC/embedded)
? RT performance > 1 Hz

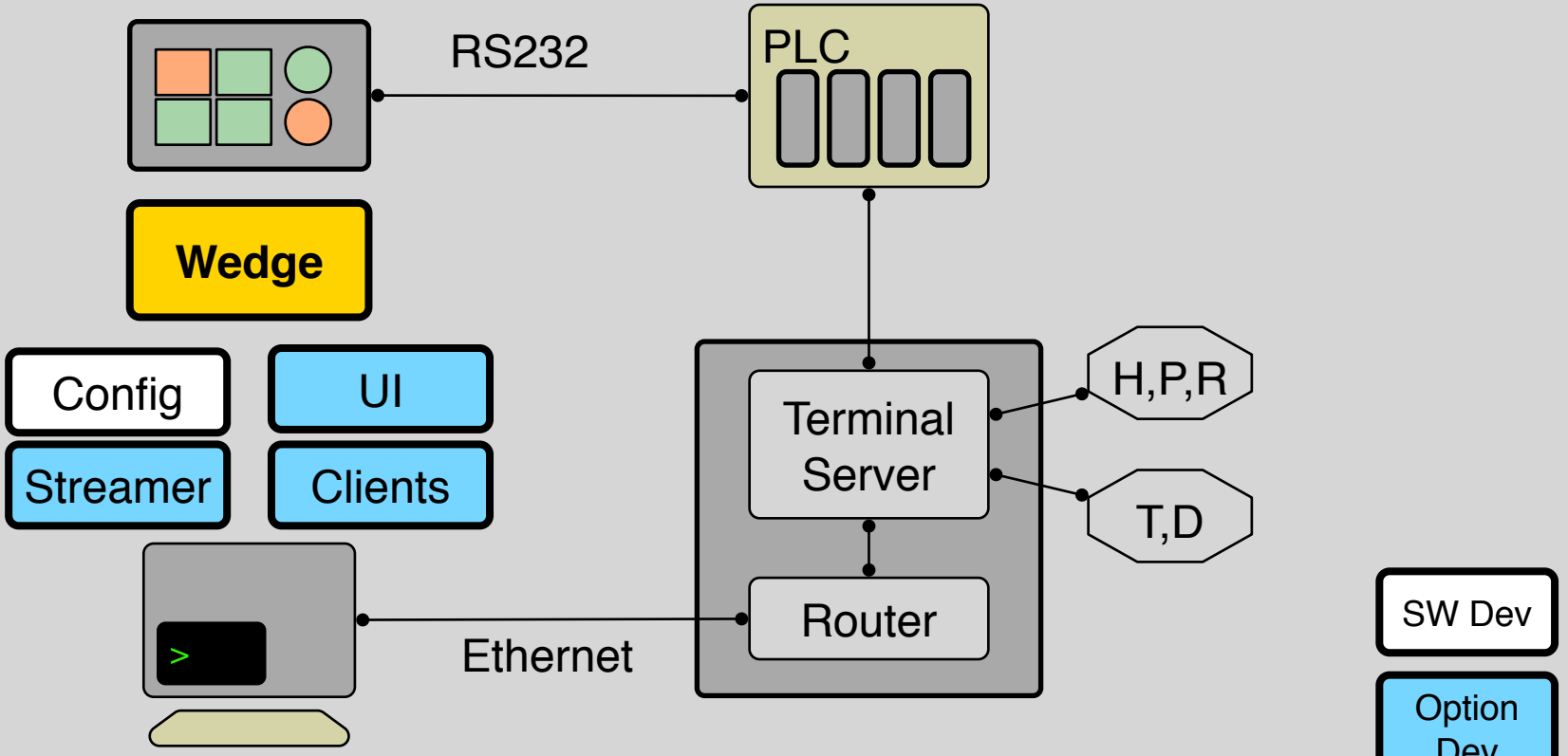


Use COTS SW-only solution to minimize SW/HW development

Concept: COTS Software

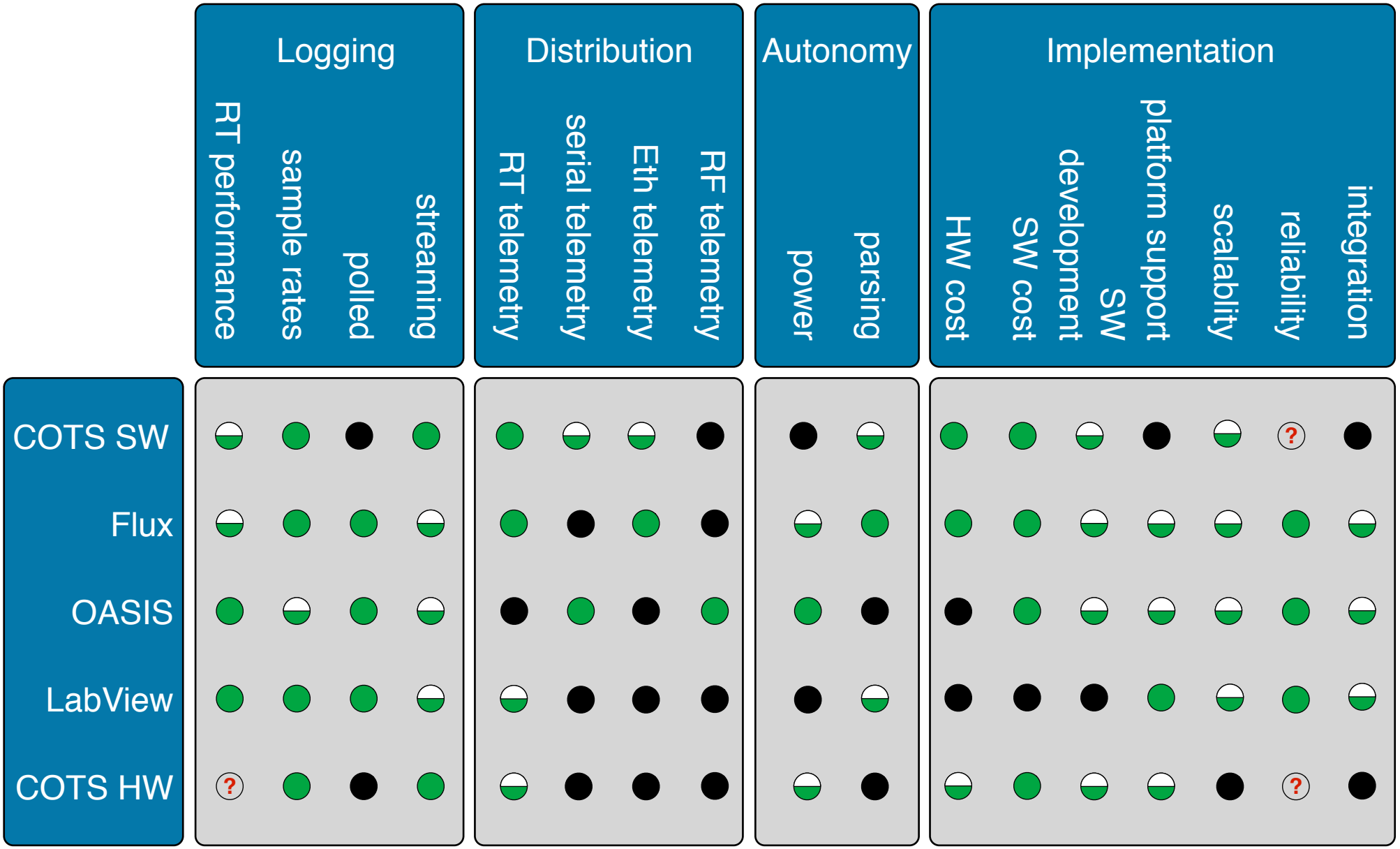
Approach: Software Solution

+ Fast, low cost dev
+ RT distribution support
+ polled or streaming
+ Easy to use
+ parsing, aggregation
- Not open source
- Autonomous use fit
? RT performance



Software Concept Trades

Details



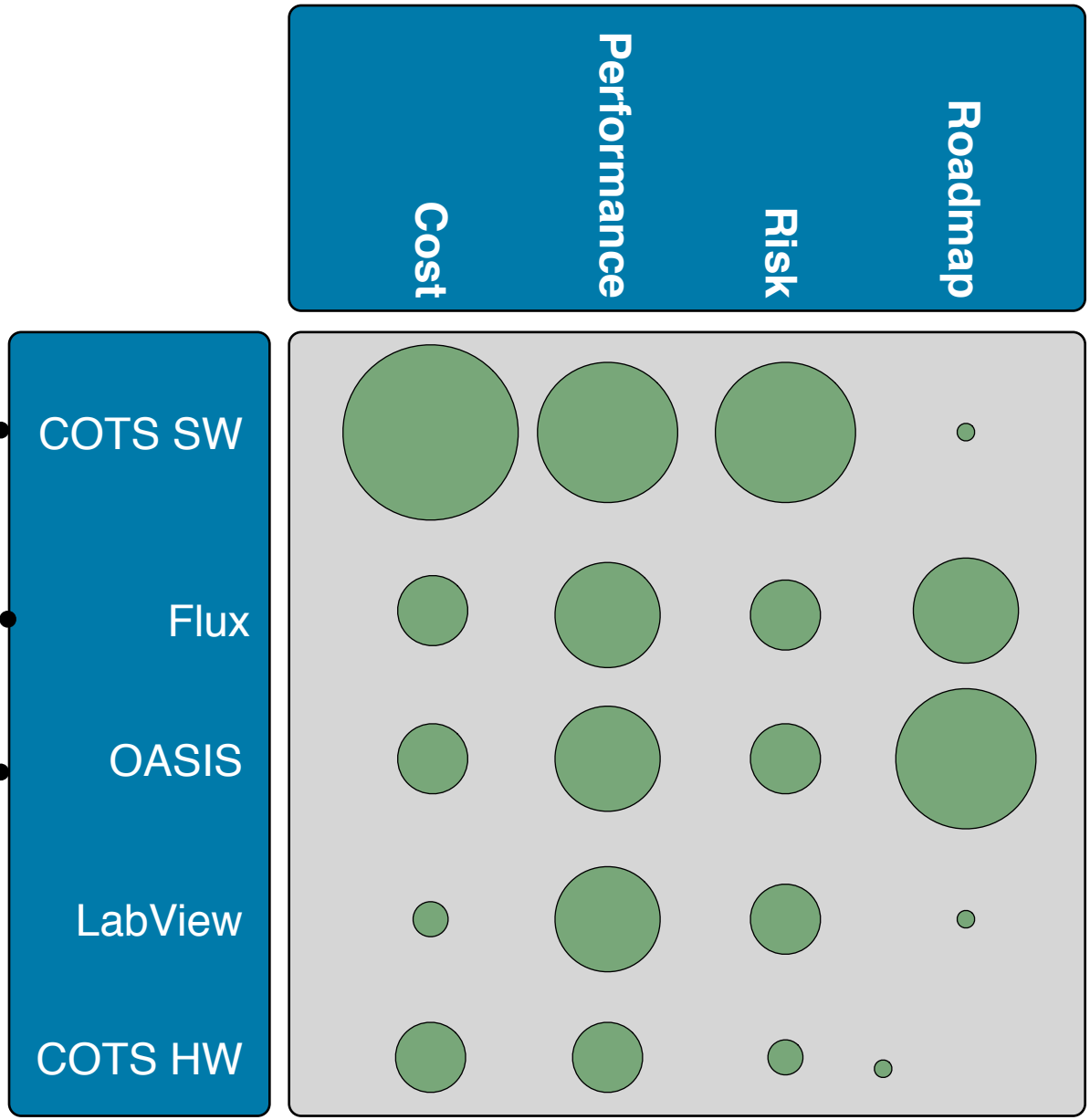
Relative Advantage

[bigger : better]

- Low Cost
- Easy to use
- Best Option for TNC

- Designed for distributed or autonomous use
- Moderate power

- Reliable, low power
- Integration effort required for tethered use




Discussion