Situational Awareness Architectural Patterns

Mike Gagliardi, Bill Wood, Len Bass – SEI
Manuel Beltran – Boeing

For additional Info: wgw@sei.cmu.edu

Motivation

- *Software Patterns* are the codification of common problems within a domain using a template
  - problem, context, rules for solution, examples
- *Pattern Languages* describe the relationships between patterns - straightforward
- Patterns have become prominent in Software Development
- *Templates* had been proposed for both System and System-of-Systems (SoS) patterns
  - Cloutier (Stevens Institute), Open Systems Group

Getting Started

- Primary interest is in SoS
- Common functionality to focus efforts
  - Situational Awareness (SA)
- Chose SoS *Pattern Template*
  - Added Quality Attributes (performance, availability, security)
- Developed a *Questionnaire* (somewhat SA specific)
- Conducted 4 interviews (1 was a follow-on)

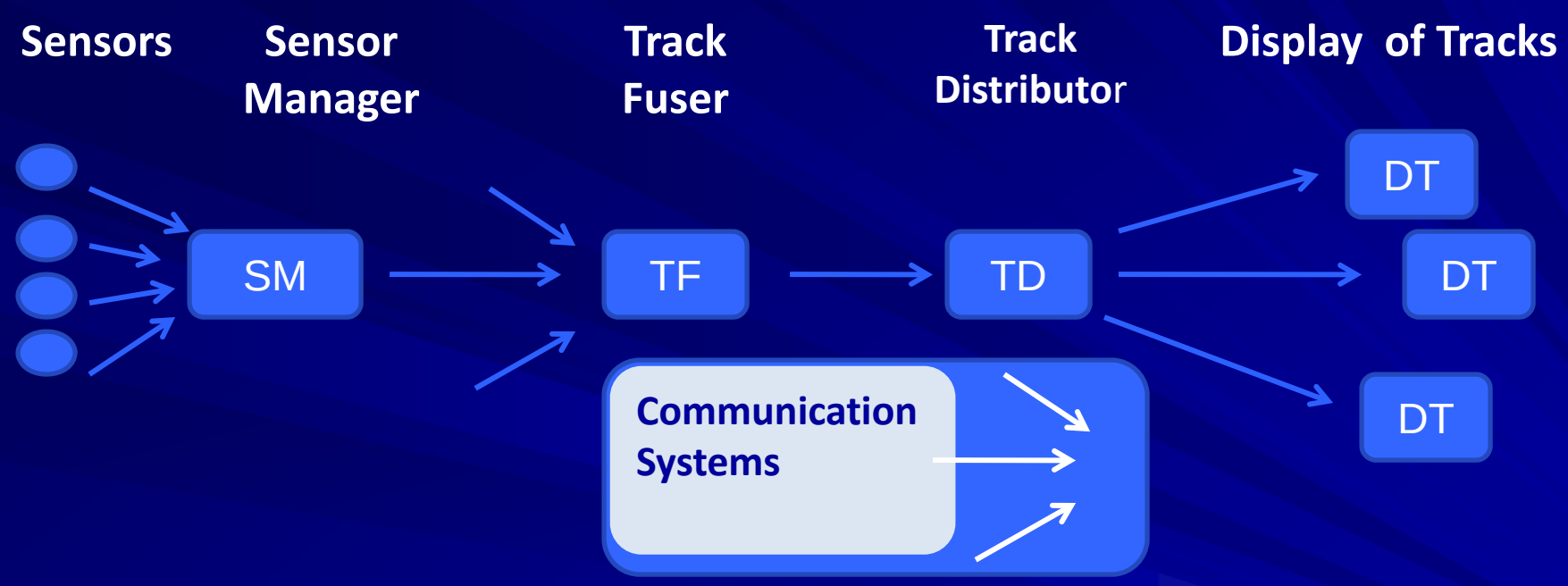
Pattern Template

	Pattern Name
1	Aliases
2	Keywords
3	Problem Context
4	Problem Description
5	Forces (**Quality Attributes**)
6	Pattern Solution
7	Diagram
8	Interfaces
9	Resulting Context
10	Example
11	Pattern Rationale
12	Known Uses
13	Related Patterns
14	References

Analysis

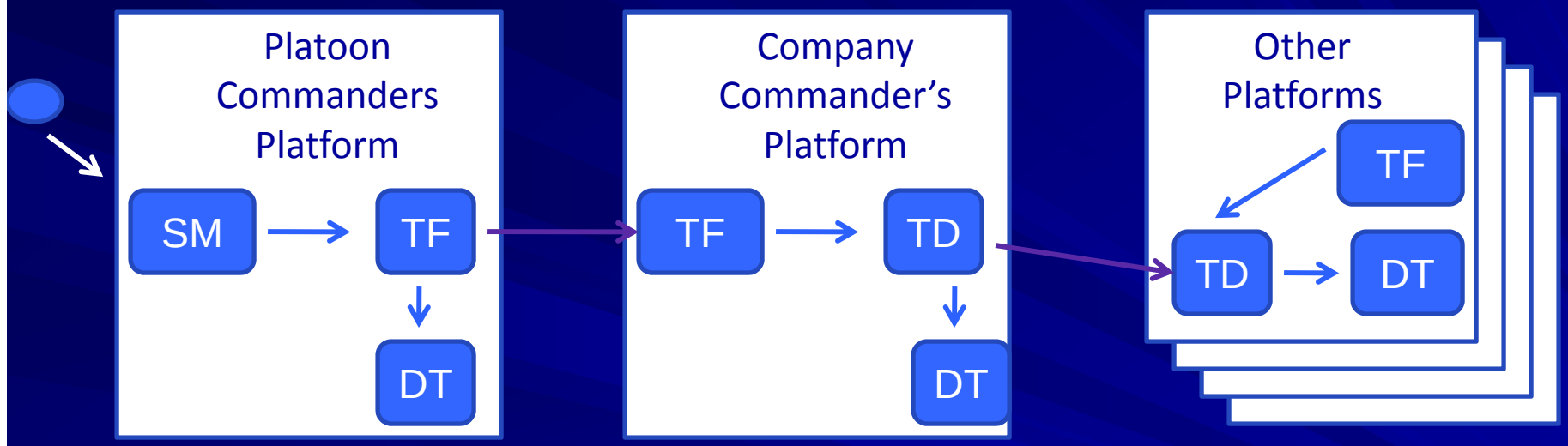
- Extracted data from the interviews into individual 5 SoS patterns (functional)
  - Capture the essentials of Quality Attributes and variants in the individual patterns(functional)
  - Use Mission Threads to inter-relate pattern functionality
  - Extract Quality Attribute requirements from steps in the Mission Thread
  - Use a pattern language to select variants to support architects

Situational Awareness Patterns



- Doctrine for displaying track data by hierarchy and geography
- LoS Radio communication is low-bandwidth and subject to splits and merges
- Signal Corp. develops multi-cast channels based on force structure, geography, intelligence and enemy posture

Deployment Example



SM: Sensor Manager
TF : Track Fuser
TD : Track Distributor
DT : Display Track

→ P2P Wireless Communication

→ Intra Platform Communications

→ LoS Radio Communications

Variants

- GPS sensor is treated differently from other sensors
- Fusion of tracks can be done
  - each platform
  - SoS wide basis
  - **multi-regional basis**

Context

- Command and Control
  - Brigade and below
  - Manned and unmanned vehicles, unmanned sensors, dismounted soldiers
  - Terrain map, streets, buildings, bridges etc
  - Own, enemy, coalition forces, non-combatants
- **Fusion**
  - Each platform has multiple sensors, each developing sensor tracks
  - Each platform fuses tracks from all own sensors forms platform tracks
  - Options on cross-platform track formation
    - Send all tracks to everyone and fuse locally (high bandwidth, simple discovery)
    - Send tracks to hierarchical and geographical neighbors (low bandwidth, complex discovery)
    - Send all tracks to single brigade fusion engine (high bandwidth, simple discovery)
    - Send to multiple regional fusion engines (low bandwidth, reasonable discovery)

Performance- Problem

Quality Attribute	Concern	Description
Performance	Timeliness	Correlating a new platform track with an existing regional track (formed from tracks from other platforms) requires a search through a number of nearby regional tracks and is compute- time sensitive.
		The RTF must fuse incoming platform tracks with existing regional tracks and create new regional tracks as necessary. The processing requirements to do this will depend on the number of sensed EoIs in the region and the number of platforms that sense these EoIs and send platform tracks to the RTF- the *workload* for the RTF.
	Bandwidth	Sending platform tracks to the RTF should have a predictable, limited communication bandwidth usage.
		If a platform track is updated based on multiple asynchronous updates from different sensors, then sending after each update increases bandwidth usage.
		Since all platform tracks are sent to the RTF platform, it should have a central position in the dynamic network structure, such that few communication “hops” are needed to move data from individual platforms to the RTF

Performance- Solution

Quality Attribute	Concern	Description
Performance	Timeliness	Each platform track has a consistent identification and the RFT remembers the historically associated regional track.
		The RTF has the capability to process a well defined workload that exceeds the maximum workload expected in the field.
	Network Bandwidth Usage	Simulation exercises have been conducted with multiple regions, and a high concentration (above expectations) of EoIs and sensor platforms. They have demonstrated that the bandwidth consumed by sending tracks to multiple regional fusion engines is acceptable.

Availability- Problem

Quality Attribute	Concern	Description
Availability	RTF Failure	When an RTF fails, the recovery could be to an RTF in the same platform or in another “standby” platform without dropping tracks.
	Splitting and Merging	When the communication network splits, the region will have to split into multiple fusion regions to accommodate the communication network split.
	Saturation	When the workload being presented to the RTF approaches its processing limitations, the workload must be decreased. This can be done by informing some of the platforms to reduce their sending frequency for some platform tracks, or by re-configuring the regions.
	Reliability of sending platform tracks	Since platform track updates are sent periodically, they need not be received reliably. However, new platform tracks must be received reliably.

Availability- Solution

Quality Attribute	Concern	Description
Availability	TF Failure	When the individual TF in a platform fails, a redundant copy will be standing by in the same platform to take over responsibility within 5 seconds. When an RTF fails, another platform was assigned as a standby RTF, and contains the necessary historical data to restart within 5 seconds. Simulation exercises demonstrated that no tracks were dropped under a heavy workload. A new standby platform is immediately initiated.
	Splitting and Merging	A multi-region simulation with a heavy workload in each region was conducted. When the network split and a region crossed both networks, the region was successfully split without dropping tracks.
	Saturation	During a multi-regional simulation, the workload in one region was cranked up till saturation of the RTF occurred. The RTF sent messages to platforms causing their frequency of sending platform tracks to be reduced, and no tracks were dropped.
	Reliability of delivery	There is no reliable protocol within the communication system, so the reliable messaging will have to be done at the application level.

Communications System

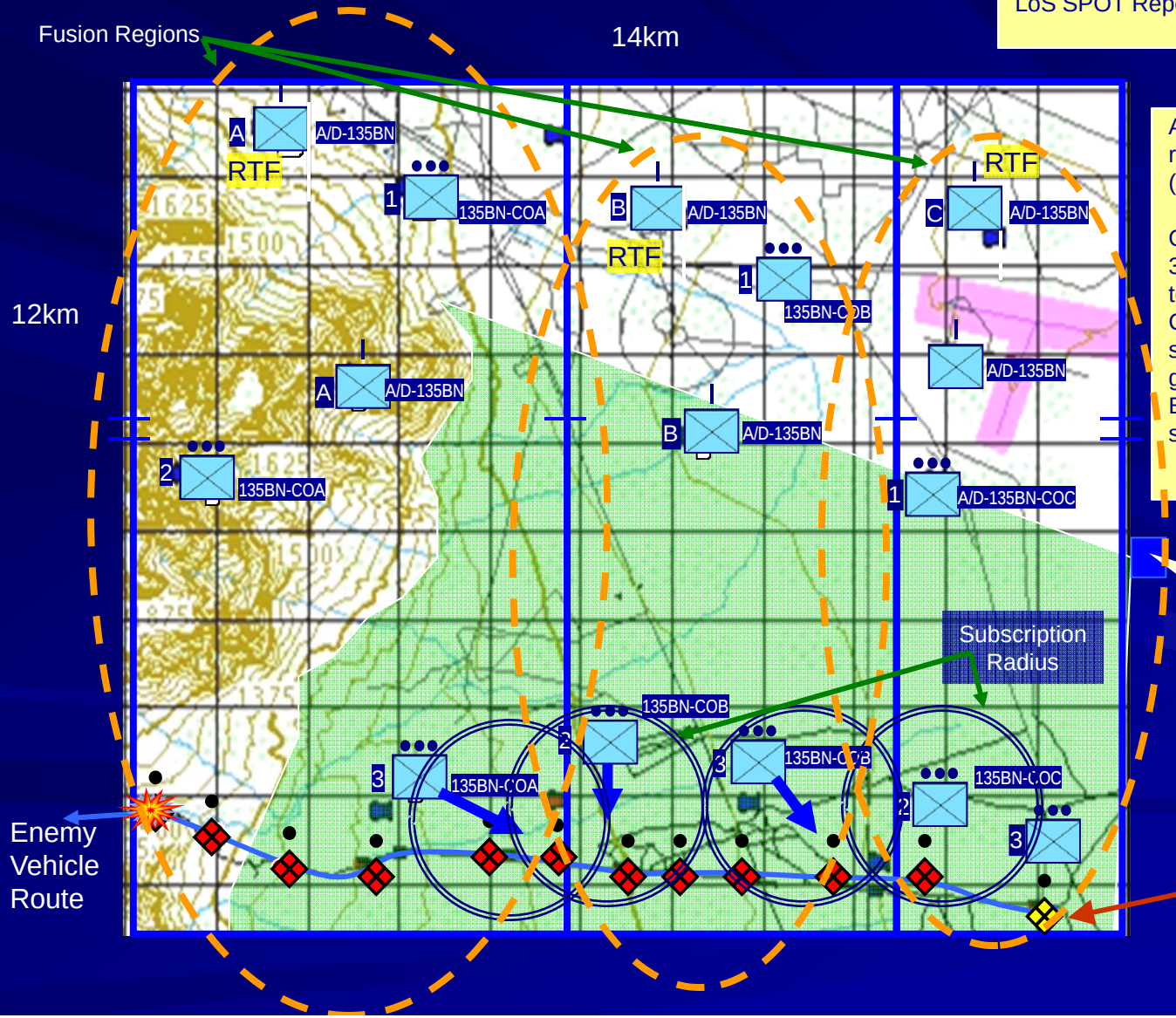
Quality Attribute	Concern	Description
Performance	Latency between Send and Receive	This depends on the number of hops that a message takes between the sender and receiver. And on the wait time in queues in hopping platforms. As the workload presented to the network increases, the wait time also increases.
	Bandwidth	The network bandwidth used depends on the number of messages sent. If a company has a multi-cast channel assigned to all members, then a message sent by a platoon member will be received by all company members; not necessarily in a single hop, since they may not all be in LoS of the sender.
	Bandwidth	The network will operate well when it is lightly loaded, but will become saturated at high traffic. When this occurs, critical messages must still get delivered with a reasonable latency, and less important messages must be suppressed and/or discarded

Mission Thread Context

- Each company has a single logical multi-cast channel connecting all platforms
- Each company and the brigade commander share a separate logical multi-cast channel
- The assignment is “defensive”, the brigade communication structure is well understood, and “discovery” is done statically

Example BN Deployment at WSMR Goat Herder Scenario with Happy Ending SA Dissemination

Initial detection takes place via Ground Radar. A SPOT Report modifies the BSO to assign affiliation and classification to the vehicle. As the vehicle moves west on the dirt road the vehicle track is transferred to adjacent fusion region. Once vehicle passes beyond radar sensor LoS SPOT Reports maintain the track on the vehicle.



Assume deployment is Mountainous Terrain resulting in a Subscription Radius of 1.5 Km (Urban Canyon).

Once initial SPOT report is created, Co B 3rd PL, Co B 2nd PL, and Co A 3rd PL move to intercept Goat Herder truck. This places Co C 2nd PL and Co B 3rd PL with subscription radius, resulting in geographical dissemination, and places Co B 2nd PL and Co A 3rd within geographical subscription radius.

Ground Radar Sensor providing detections of moving vehicles

Extractions from the Mission Thread

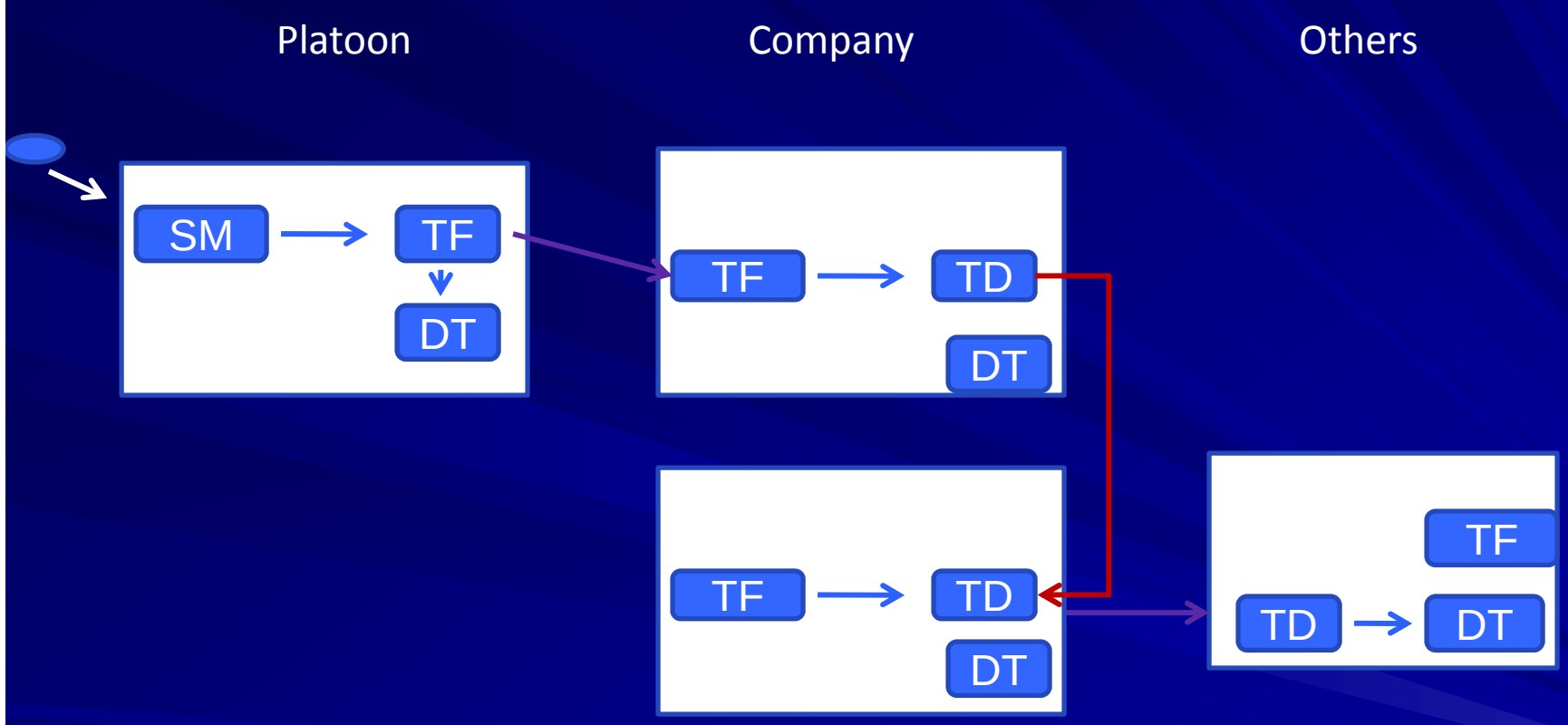
Performance

- The track is marked suspicious (yellow) or hostile (red) by a platoon vehicle and distributed both hierarchically and geographically ($7km^2$) within 10 seconds
  - Worst case Compute + Communication < 10 seconds
- Radar ground tracks are distributed across the battalion from a single source and distributed hierachically ($168km^2$) within 1 minute.
  - $(SM+TF)_{Radar}+(TF+RTF+TD)_{CoC}+(TD+RTF+TD)_{CoB}+(TD+RTF+TD)_{CoA}+(TD+TF+DT)_{3rdPL}+4*T_{los}+4*T_i < 1$ minute

Availability

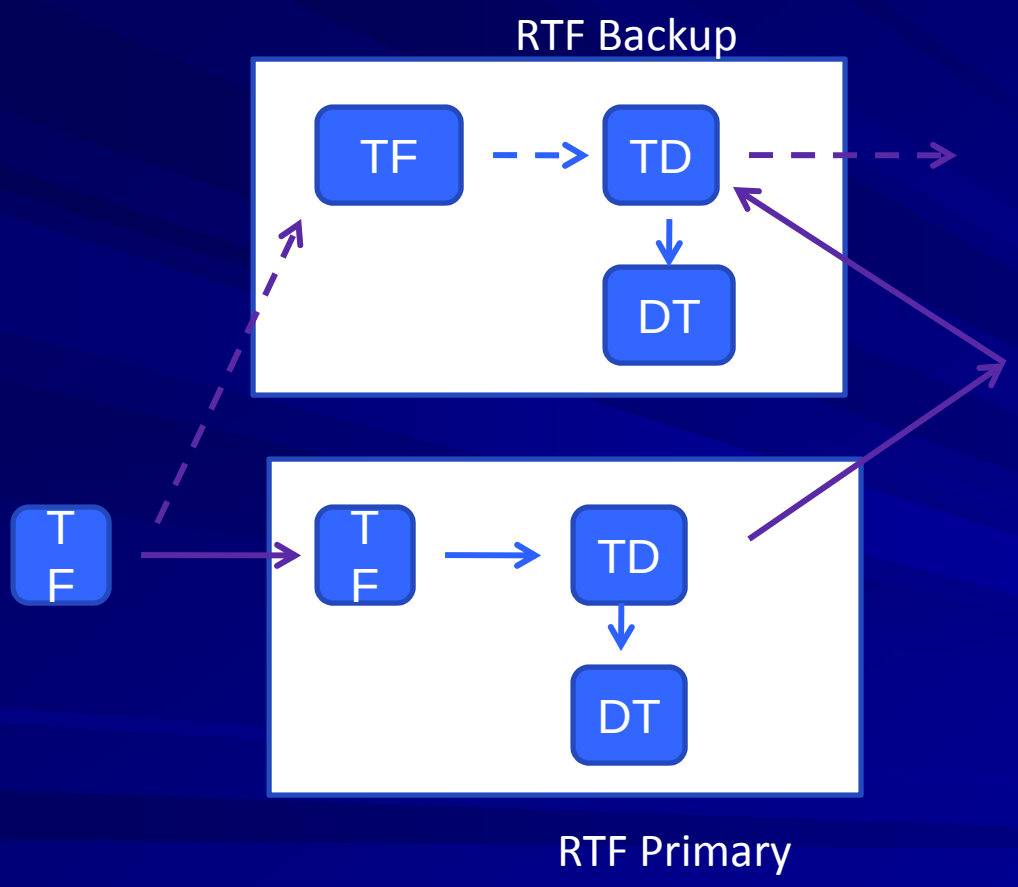
- Each company command post (XO) serves as the RTF for a region. The company commander platform (CDR) is in standby to take over and has all the timley updates.

Pattern Language



$$Tp2p + Tsm + 2 * Ttf + 3 * Ttd + Tdt + 3 * Ti + 3 * Tlos \leq 10$$

Pattern Language- Availability



Conclusion

- It's a lot of work to establish the patterns from interviews and fill in the templates
- Deciding what should be in the patterns and what should be in the language is easy to say and hard to do
- The language can guide the architect, but leaves a complicated design for him to do
- Having patterns is better than starting from scratch

Further Thoughts

- The SoS architect generates alternative high level designs and then chooses between them based on some measures of how they satisfy the architectural drivers (requirements, quality attributes)
- The system capabilities involved in the alternatives are large scale and are distributed across different platforms.
- The relationships between these capabilities are the focus of the SoS architect.

Capabilities

Track Fusion

- Fuse tracks of same environmental object from different sensors

Track Distribution

- Send tracks appropriately between platforms
- Take advantage of communication structure

Track Display

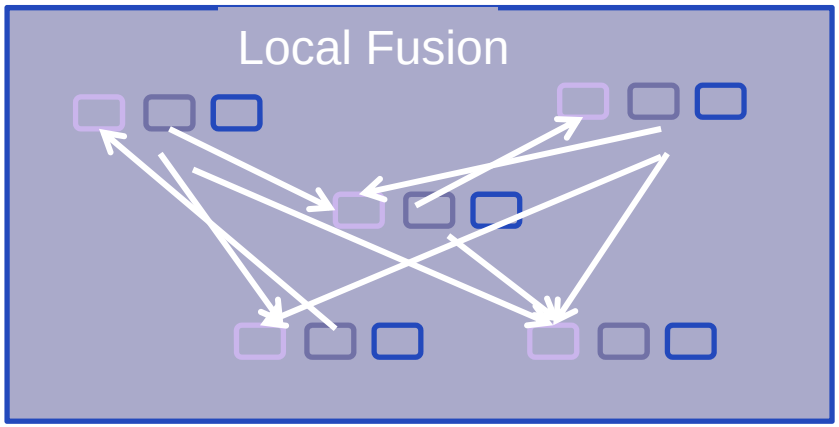
- Satisfy doctrine- hierarchical and geographical display of tracks
- Common Identifier for tracks of same object on different platforms

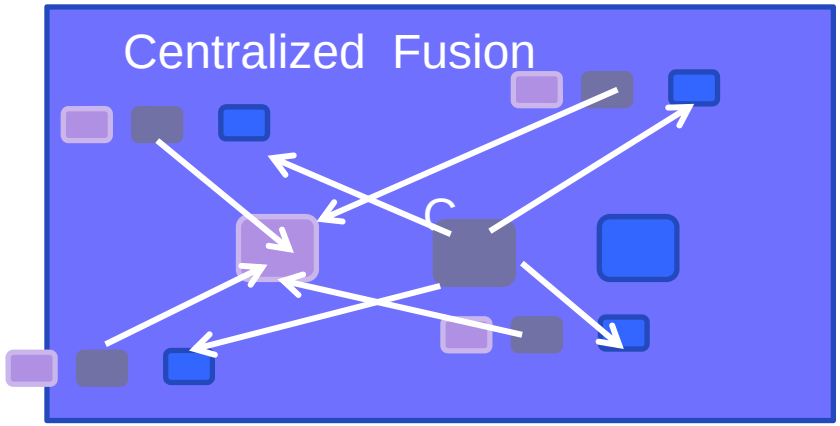
SoS Architects Alternatives

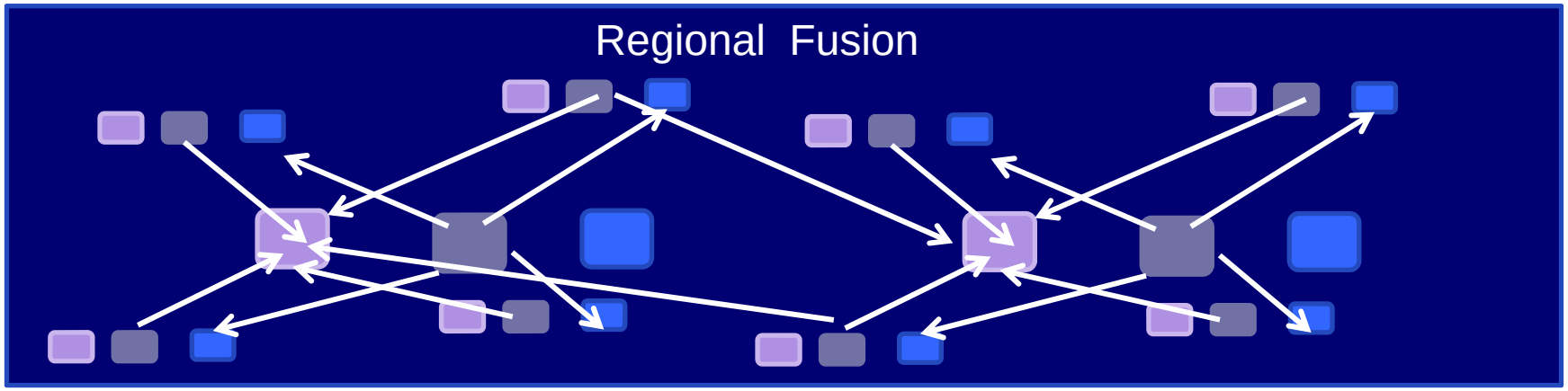
Fusion

Distribution

Display







Important Capabilities

#	Capability	Local	Central	Region
1	Network Bandwidth used	Highest (800)	Middle (720)	Lowest (640)
2	Unique Identifier	Highest	Lowest	Middle
3	Remote Display Latency	Lowest	Highest	Middle
4	TF Failure- No dropped Tracks	Lowest	Lowest	Middle
5	Hierarchical Display	Same	Same	Same
6	Geographical Display	Highest	Lowest	Middle
7	Network Split	Easiest	Highest	Middle