# Mission-based Architecture for Swarm Composability (MASC)

By
CDR Katy Giles, USN
5th Annual SERC Doctoral Students Forum
November 7, 2017
FHI 360 CONFERENCE CENTER
1825 Connecticut Avenue NW
8th Floor
Washington, DC 20009

www.sercuarc.org

# Agenda

- Background
- Research Focus
- MASC Architecture
- MASC Methodology
- Conclusions and Future Work

# Background- What is a swarm system?

*"Swarm robotics is the study of how large numbers of relatively simply physically embodied agents can be designed such that a desired collective behavior emerges from the local interaction among agents and between agents and the environment."* [1]

- General attributes:
  - Decentralized control
  - Agent autonomy
  - Large numbers of agents following simple rules
- Relation to systems engineering:
  - Swarm systems are complex adaptive systems
  - Exhibit collective emergent behavior
  - INCOSE complex systems guiding principles[2]:
    - Identify patterns
    - "Influence & intervene" rather than control
    - "Zoom in and zoom out," multiple views
  - Cognitively challenging to operate multiple vehicles[3,4,5]
    - Air traffic controller research

www.wired.com

www.wikipedia.org



Reference: [1] Sahin 2005, [2] Sheard 2015, [3] Cummings 2007, [4] Olson 2004, [5] Hillburn 1997

# Motivation

## Current

**Field**

**Programmer**

www.modelairplanenews.com

**Single Vehicle Pilot**

## Future

www.af.mil

**Fleet**

www.popsci.com

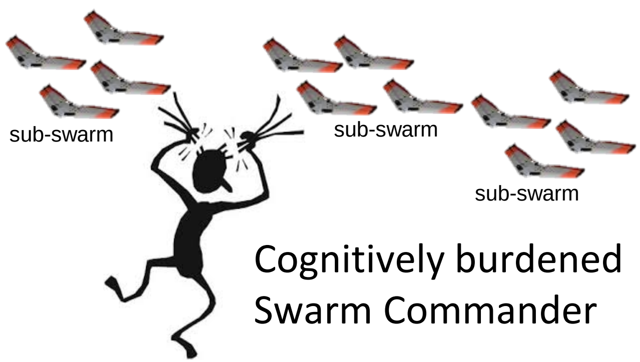
**Operator**

**Swarm Commander**

# Research Focus

## Problem

- Informal relationship between swarm mission engineering and swarm systems engineering impedes **architecture reusability**
- Swarm system architecture is dominated by bottom-up, behavior-based design




- Informal
- Operated at single behavior level
- Different action plans for each mission
- Low flexibility
- Micro-management approach

## Proposed Solution

Transfer typical rule-based decisions from the Swarm Commander to the swarm, freeing the human to make **rules of engagement** related decisions

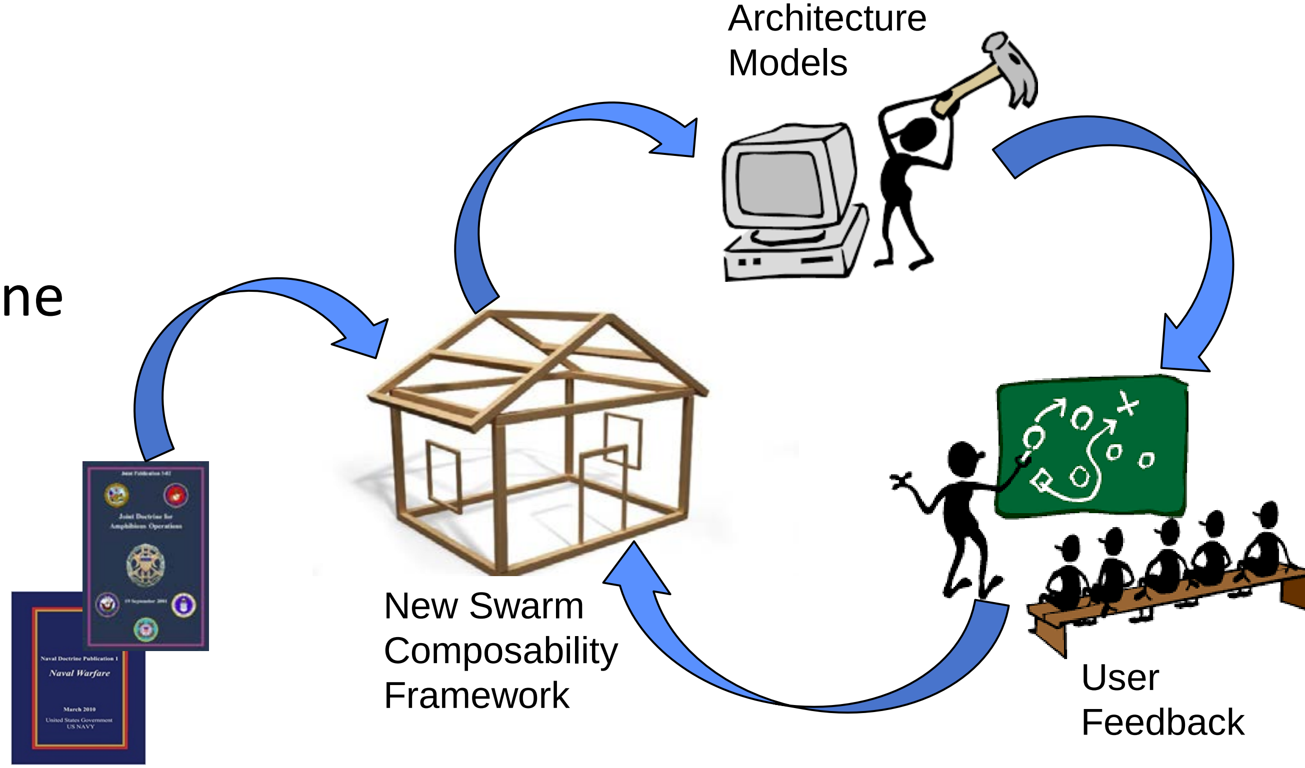

- Formal
- Reusable common patterns
- Modular
- Intuitive
- Platform agnostic
- Experiential heuristics-based

*Swarm systems are being engineered without guidance from swarm doctrine*

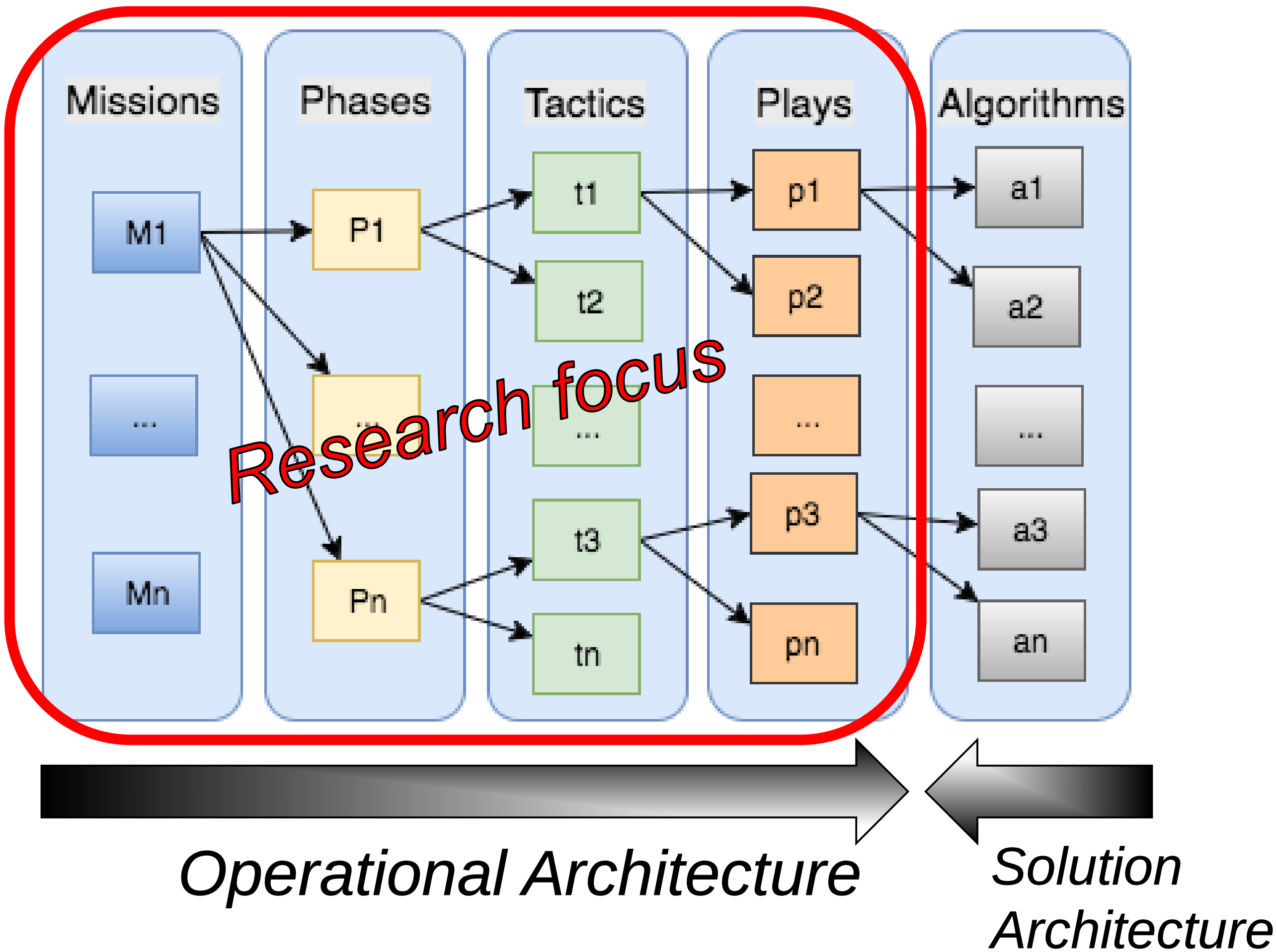
# Intended Benefits of Swarm Architecture

Formalize relationship between swarm mission engineering and swarm systems engineering to promote architecture reusability

- Intuitiveness
- Modularity
- Composability
- Mission Doctrine Integration



# Mission-based Architecture for Swarm Composability



# Swarm Mission

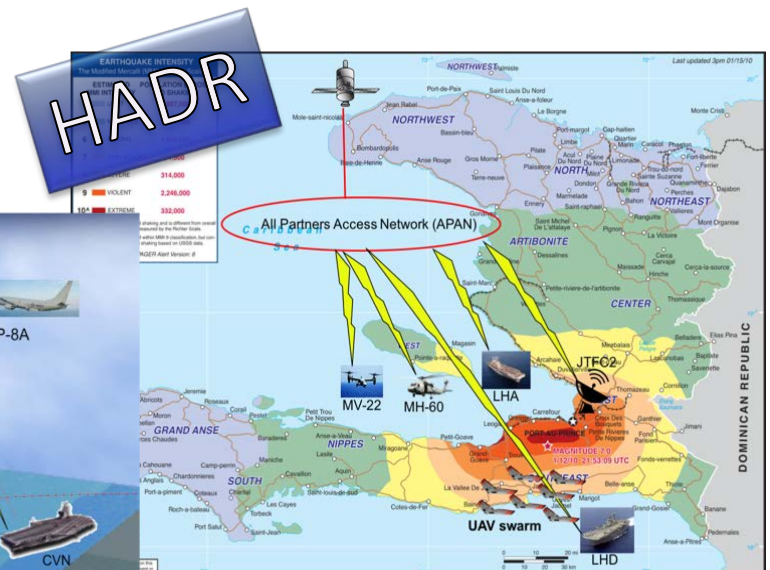


- *Swarm mission* describes the overall task and purpose delineating actions assigned to the UAV swarm
  - —Examples: intelligence, surveillance, reconnaissance (ISR), humanitarian assistance/disaster relief (HADR), search and rescue (SAR), and counter drug operations

- Research focuses on three basic missions:


Source: ARSENL 2015


Adapted from: Okon 2012

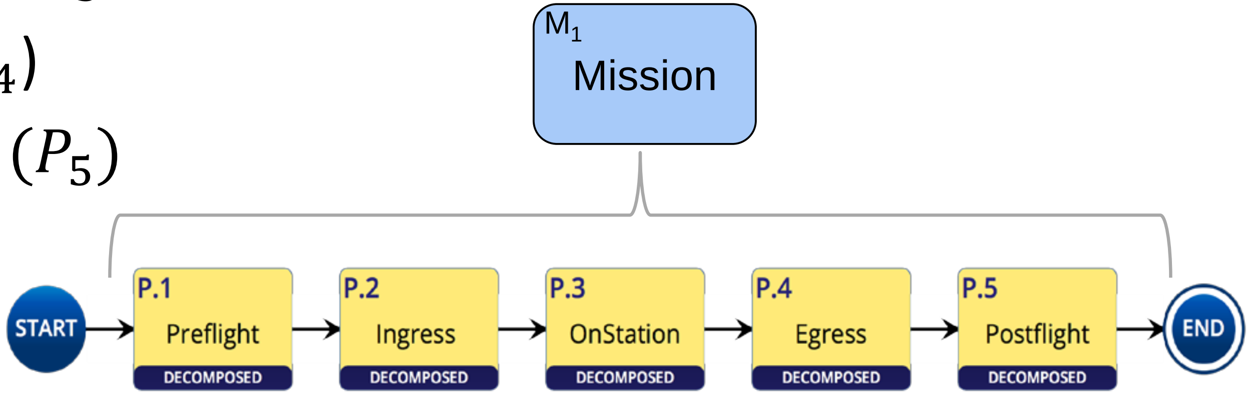



Adapted from: USAID 2010

# Swarm Phase



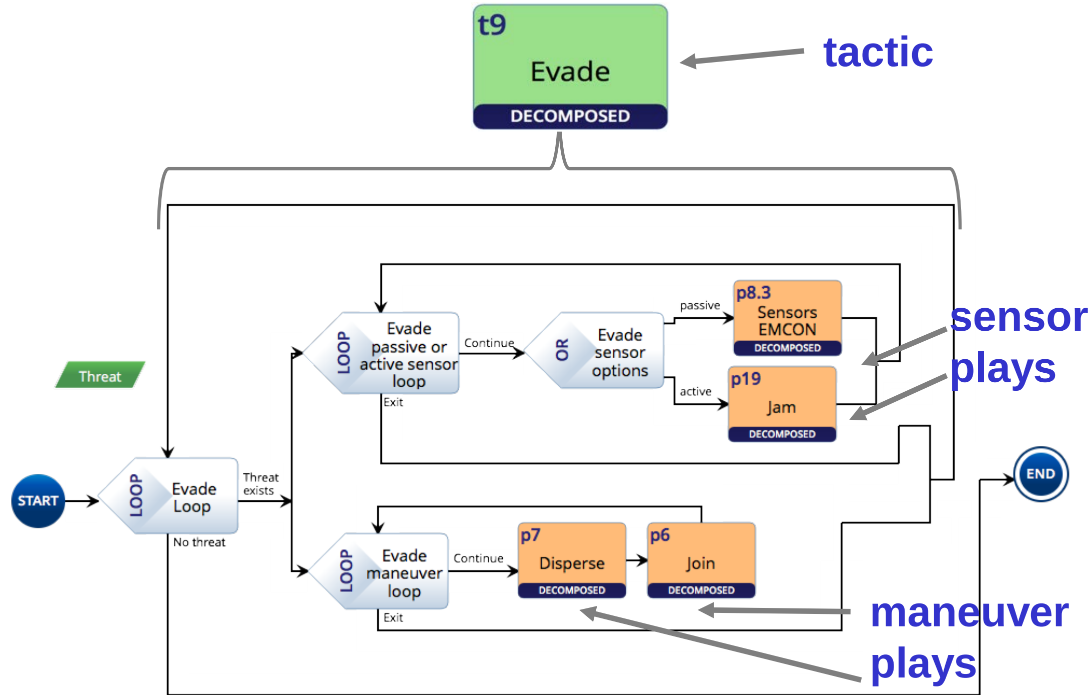
- *Swarm phase* describes a distinct time period within the mission
- There are five operational phases in a swarm mission ($M$):
  - Preflight ($P_1$)
  - Ingress ($P_2$)
  - OnStation ($P_3$)
  - Egress ($P_4$)
  - Postflight ($P_5$)




Top-level diagram of simulation developed in *Innoslate*™

# Swarm Tactic and Swarm Play

- *Swarm tactic:* employment and ordered arrangement of agents in relation to one another for the purpose of performing a specific task
  - Each tactic composed of one or more swarm sensor and maneuver plays
  - Designed to be used in multiple missions
  - Examples: search, divide, evade, and attack



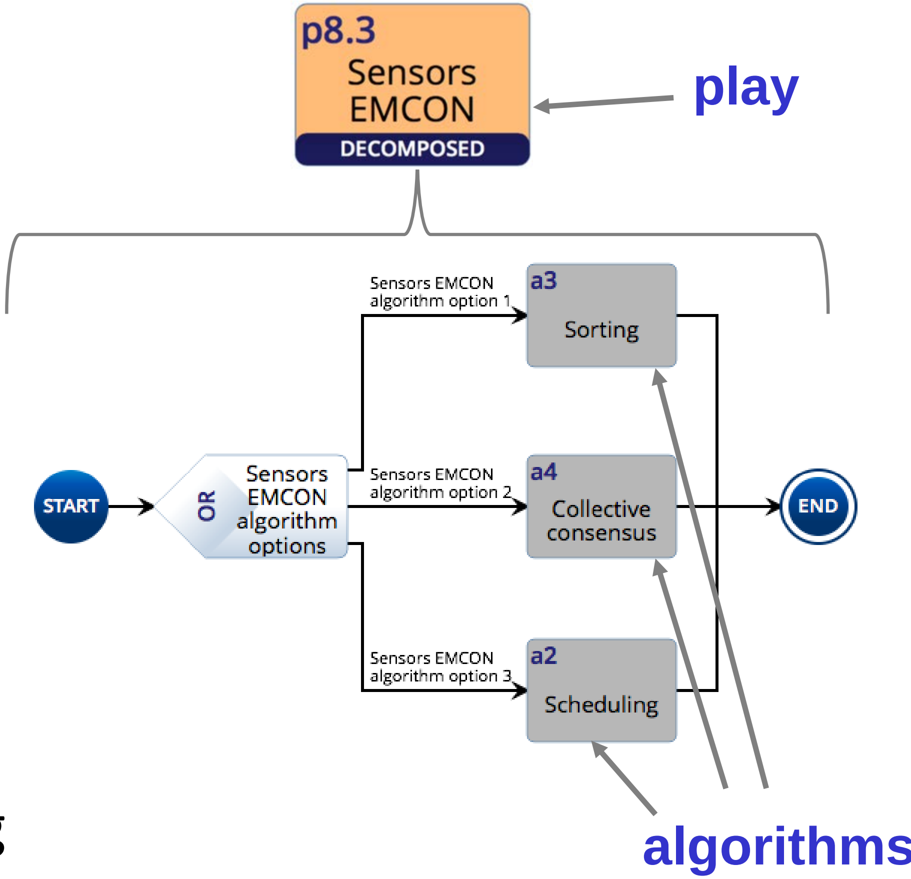

- *Swarm play:* maneuvers and behaviors of swarm as a collective of agents with specific triggers and temporal constraints
  - Each play composed of one or more swarm algorithms
  - Designed to be used in multiple missions
  - Examples: launch, transit, split, join, or bit, and sensors EMCON

Diagram of part of simulation developed in *Innoslate*™

# Swarm Algorithm



- Three general categories[1]:
  - Reactive: sense and act, pheromone-based, and other biologically inspired algorithms
    - Reynold's "Boids" flocking, bee colony, ant colony
  - Deliberative: require information trading and solution deliberations
    - Sorting, consensus, greedy selection, physicomimetic
  - Evolutionary: genetic algorithms and other fitness-based optimization functions
- *Swarm algorithms:* step-by-step procedures used by the controlling software to solve a recurrent task






References: [1]Senanayake et al. 2015, Mitchell 2009

# MASC Methodology

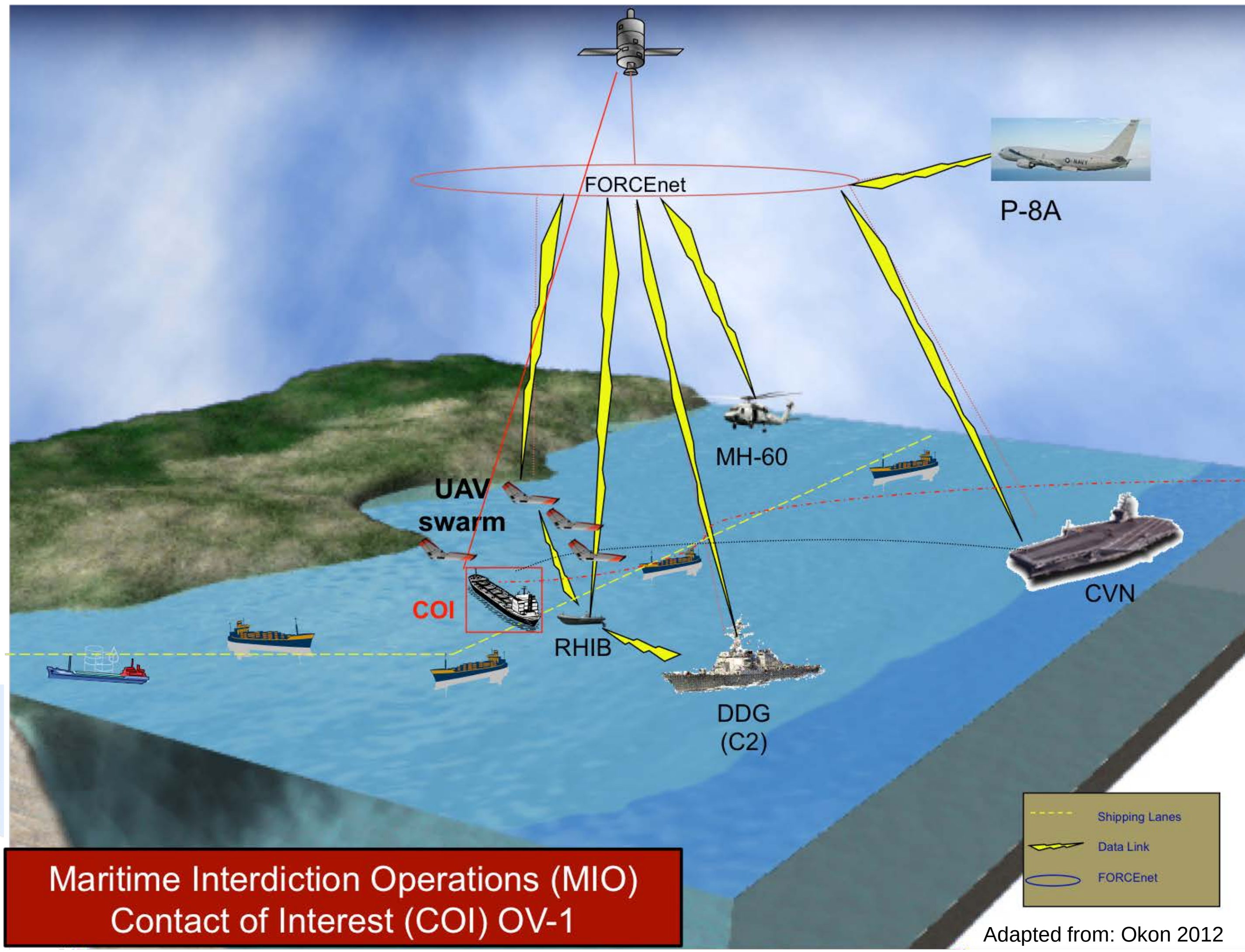
*mission doctrine*

1. Develop mission scenario
2. Depict swarm behavior at tactics level
3. Develop mission simulation beginning at phase level
4. Check for logical errors
5. Review implementation with stakeholder
6. Revise tactics
7. Swarm doctrine & swarm system requirements

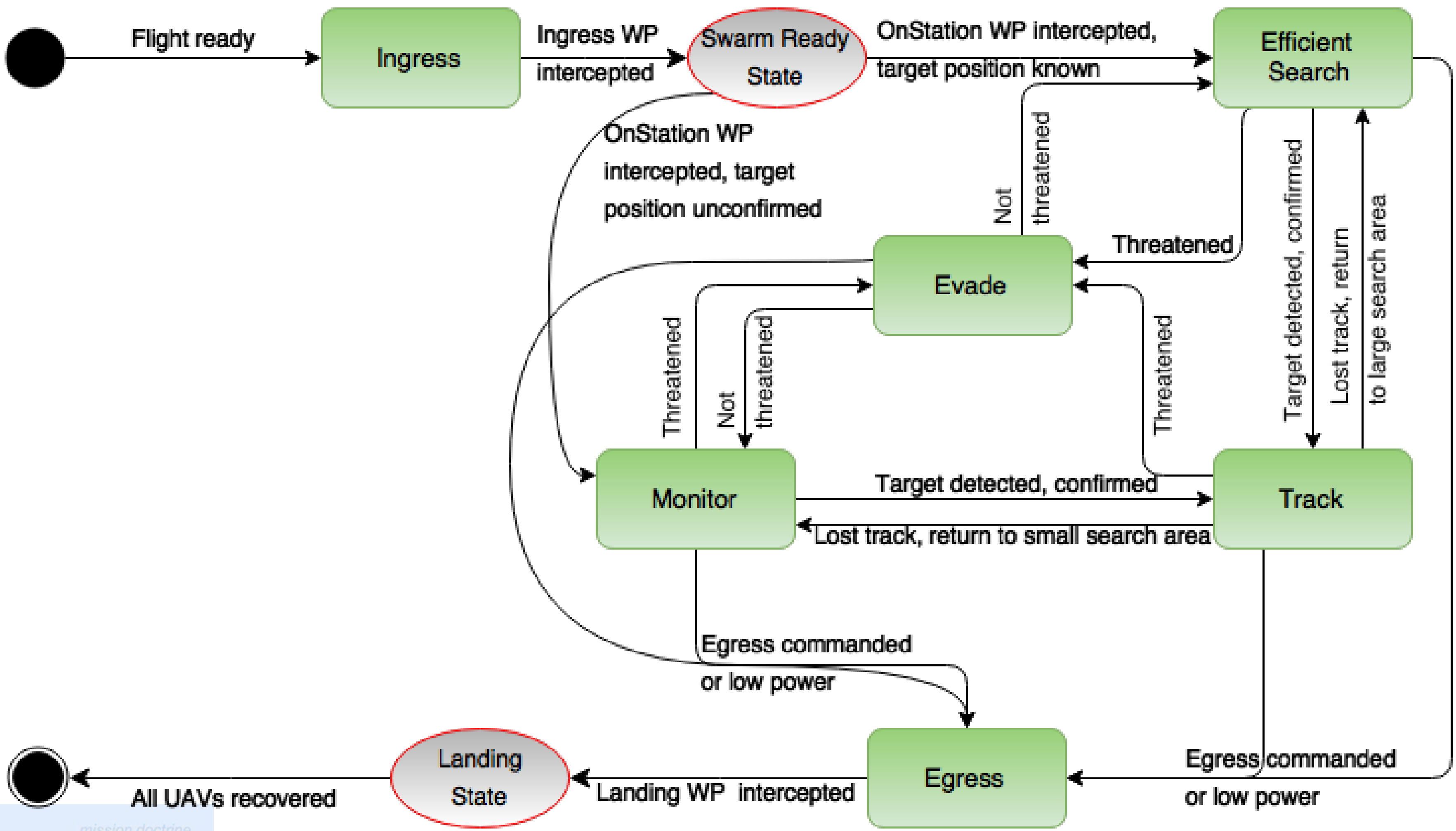
*system architecture*

- Multi-national maritime interdiction operation
- UAV swarm supports boarding team with surveillance, communication relay
- Swarm provides real-time, close range sensor collection

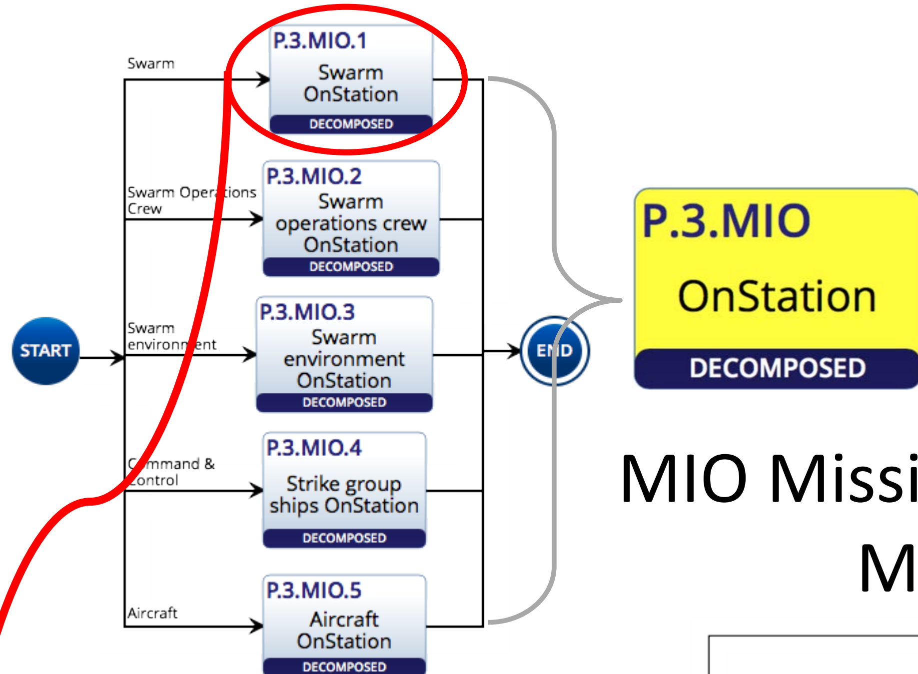
## Consider this scenario....

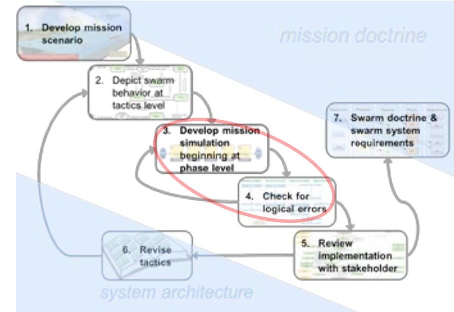



# MIO Mission Tactics Level as FSM




# MIO Mission Activity Diagram Simulation




MIO Mission at tactics level using MASC framework

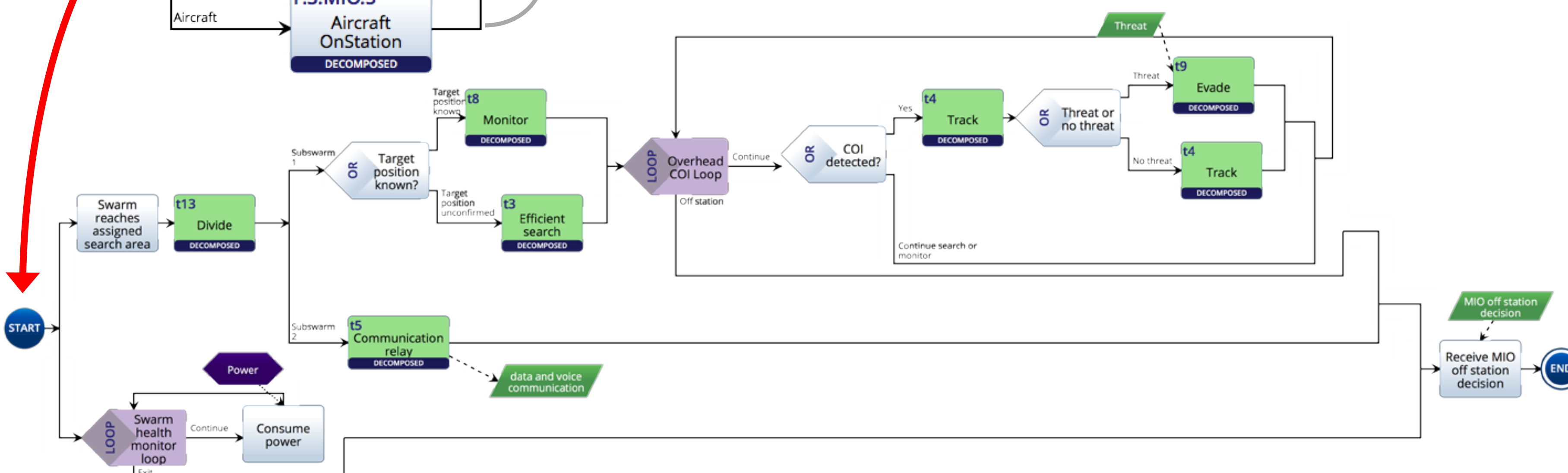
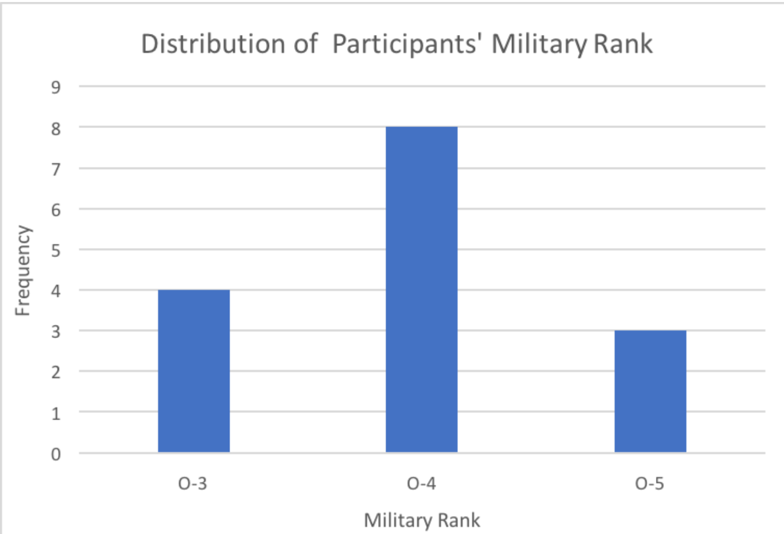
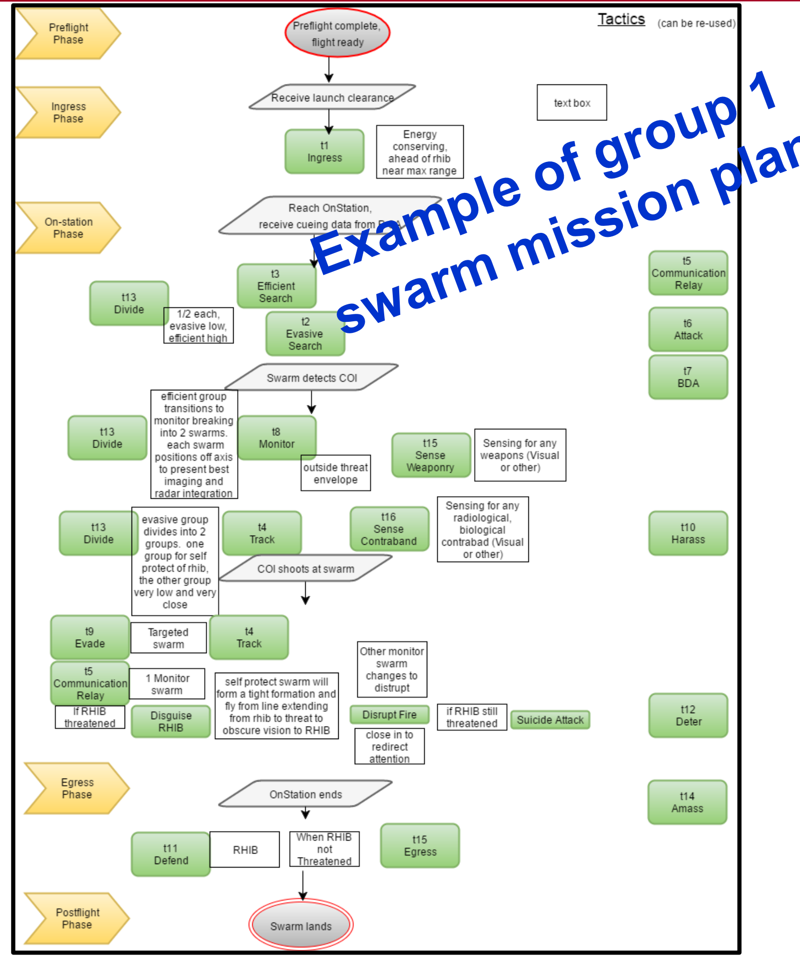




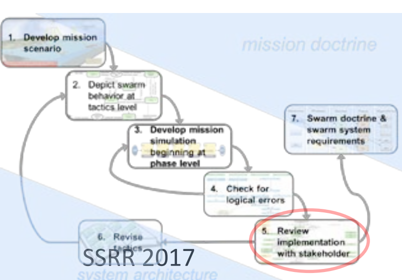

Diagram of part of simulation developed in *Innoslate*™

# Human Subjects Research-Stakeholder Feedback




Data were collected from 15 subject matter expert naval aviators and naval flight officers




Participants read the fictional MIO scenario, constructed a **UAV swarm mission** plan, and answered a survey

- Group 1 used tactics
- Group 2 used only plays, no knowledge of tactics

# Conclusions

## Modular

| Tactic | SvS | MIO | MIO HSR | HADR |
|---|---|---|---|---|
| Ingress | B | B | 7 | B |
| Evasive search | B | | 4 | |
| Efficient search | | B | 5 | B |
| Track | B | B | 7 | B |
| Communication relay | | B | 7 | B |
| Attack | B | | | |
| BDA | B | | 3 | [illegible] |
| Monitor | | B | [illegible] | [illegible] |
| Evade | [illegible] | [illegible] | [illegible] | [illegible] |
| Harass | [illegible] | [illegible] | [illegible] | [illegible] |
| Defend | [illegible] | [illegible] | [illegible] | [illegible] |
| Deter | [illegible] | [illegible] | 2 | |
| Divide | [illegible] | B | 7 | B |
| Amass | | B | 6 | B |
| Egress | B | B | 6 | B |
| ACM | | | | |
| Option | | | 1 | |

| Play | SvS | MIO | MIO HSR | HADR |
|---|---|---|---|---|
| Launch | B | B | 8 | B |
| Transit to WP | B | B | 8 | B |
| Orbit | | B | 7 | B |
| Racetrack | | B | 4 | B |
| Split (logic based) | | B | 7 | [illegible] |
| Join | B | B | [illegible] | [illegible] |
| Disperse | B | [illegible] | [illegible] | [illegible] |
| Sensors ON | [illegible] | [illegible] | [illegible] | [illegible] |
| Sensors OFF | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| [illegible] | [illegible] | [illegible] | [illegible] | B |
| [illegible] | [illegible] | [illegible] | 7 | B |
| [illegible] | [illegible] | B | 8 | B |
| [illegible] | [illegible] | B | 2 | B |
| [illegible] square pattern | B | B | 2 | B |
| Constricting square pattern | | B | 2 | B |
| Grid pattern | | B | 2 | B |
| Random pattern | B | | 3 | |
| Weapon armed | B | | | |
| Weapon fire | B | | 1 | |
| Follow target | | B | 5 | B |
| Forward communication | | B | 4 | B |
| Jam | | | 1 | |
| Smart greedy shooter | B | | 1 | |
| Patrol box shooter | B | | | |
| Wingman shooter | B | | | |
| Tail following | B | | | |
| Option | | | 5 | |

**3 mission areas, one playbook**

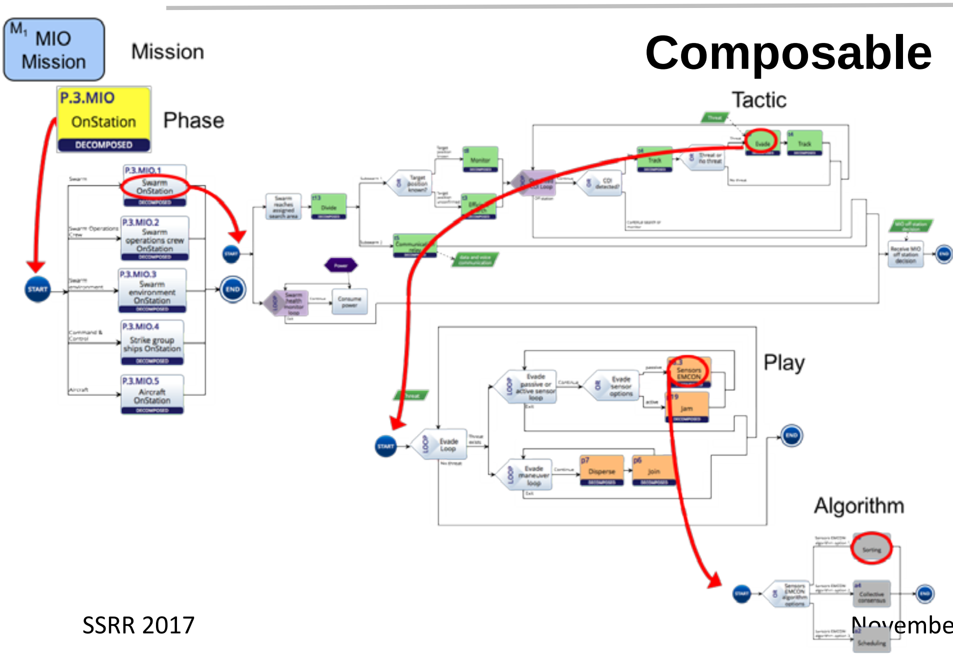
## Intuitive

*"Seemed to work well and I was able to perform the task in a timely manner.*

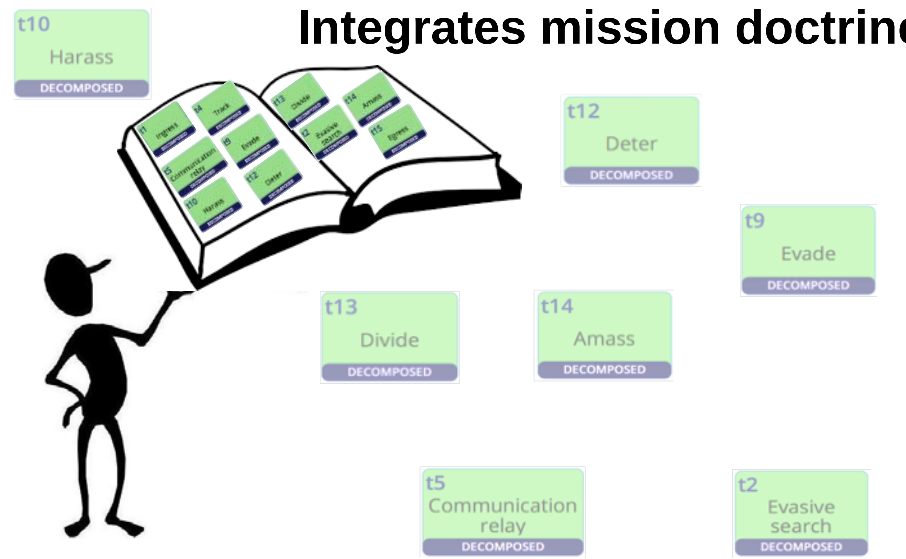
*"Playbook provided all the necessary support for this mission type"*

*"The structure of mission phases supports the mission execution"*

## Composable



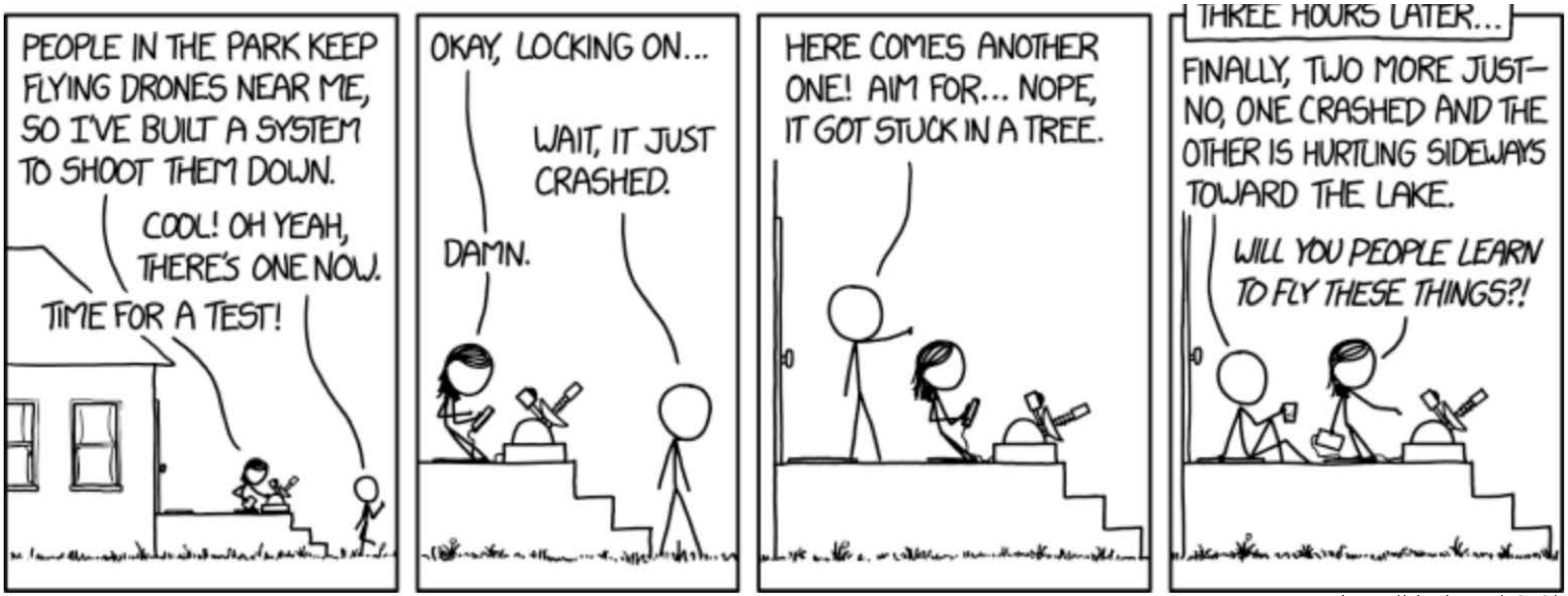

## Integrates mission doctrine



# Future Work

- Support improved graphical user interface for UAV swarm operations
- Incorporate system and operational failure modes into simulation
- Develop swarm system evaluation measures of performance

## Drone problems….




https://xkcd.com/1846/

# References


1. U.S. Department of Defense. 2013. “Unmanned Systems Integrated Roadmap.”
2. Finn, Anthony, Scheding, Steve. 2010. *Developments and Challenges for Autonomous Unmanned Vehicles*. Springer. doi:10.1007/978-3-642-10704-7.
3. Sahin, Erol. 2005. “Swarm Robotics: From Sources of Inspiration to Domains of Application,” no. 631.
4. Cao, Y.U., A.S. Fukunaga, A.B. Kahng, and F. Meng. 1997. “Cooperative Mobile Robotics: Antecedents and Directions.” *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots* 23: 226–34. doi:10.1109/IROS.1995.525801.
5. Beni, Gerardo, and Jing Wang. 1993. “Swarm Intelligence in Cellular Robotic Systems.” *Robots and Biological Systems: Towards a New Bionics? NATO ASI Series Volume 102*, no. 2: 703–12. doi:10.1007/978-3-642-58069-7_38.
6. Beni, G. 2005. “From Swarm Intelligence to Swarm Robotics.” *Swarm Robotics, Lecture Notes in Computer Science, 3342*, no. December. doi:10.1007/978-3-540-30552-1.
7. Frew, E.W., and T.X. Brown. 2008. “Airborne Communication Networks for Small Unmanned Aircraft Systems.” *Proceedings of the IEEE* 96 (12). doi:10.1109/JPROC.2008.2006127.
8. Chung, Timothy, Kevin Jones, Michael Day, Marianna Jones, and Michael Clement. "50 vs. 50 by 2015: Swarm vs. Swarm UAV Live-Fly Competition at the Naval Postgraduate School." *2013 Proceedings of AUSVI* (50).
9. Harriott, Caroline E, Adriane E Seiffert, Sean T Hayes, and Julie A Adams. 2014. “Biologically-Inspired Human-Swarm Interaction Metrics,” 1471–75.
10. Cummings, Mary L. 2004. “Human Supervisory Control of Swarming Networks.” *Annual Swarming: Autonomous Intelligent Networked,* 1–9. http://web.mit.edu/aeroastro/labs/halab/papers/cummingsswarm.pdf.
11. Cummings, Mary L., and Paul Mitchell. 2006. “Automated Scheduling Decision Support for Supervisory Control of Multiple UAVs.” *Journal of Aerospace Computing, Information, and Communication* 3 (6): 294–308. doi:10.2514/1.19599.
12. Cummings, M, S Bruni, S Mercier, and Pj Mitchell. 2007. “Automation Architecture for Single Operator, Multiple UAV Command and Control.” *The International C2 Journal* 1: 1–24. http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA478338.
13. Carr, Lee, Project Leader, Kristen Lambrecht, and Greg Whittier. 2003. “Unmanned Aerial Vehicle Operational Test and Evaluation Lessons Learned,” no. IDA Paper P-3821.

# References contd.


14. Director Operational Test & Evaluation. 2007. "Air Force Programs: Global Hawk High Altitude Endurance Unmanned Aerial Vehicle, RQ-4." http://www.dote.osd.mil/pub/reports/FY2007/pdf/other/2007DOTEAnnualReport.pdf.
15. Director Operational Test & Evaluation. 2008. "Air Force Programs: MQ-9 Reaper Hunter Killer Armed Unmanned Aircraft System (UAS)." http://www.dote.osd.mil/pub/reports/fy2011/pdf/af/2011mq9reaperuas.pdf.
16. Director Operational Test & Evaluation. 2015. "FY15 Navy Programs: RQ-21A Blackjack Unmanned Aircraft System (UAS)." http://www.dote.osd.mil/pub/reports/FY2015/pdf/navy/2015rq21a_blackjack.pdf.
17. NATO Standardization Organization. "Allied Administrative Publication–6 (AAP-6): NATO Glossary of Terms and Definitions." (2010).
18. Arquilla, John, Ronfeldt, David. 1997. "In Athena's Camp: Preparing for Conflict in the Information Age." Rand Corporation.
19. Hart, D.M., and P.A. Craig-Hart. 2004. "Reducing Swarming Theory to Practice for UAV Control." *2004 IEEE Aerospace Conference Proceedings (IEEE Cat. No.04TH8720)* 5: 3050–63. doi:10.1109/AERO.2004.1368111.
20. Arquilla, John, Rondfeldt, David. 2000. "Swarming and the Future of Conflict." *Defense*.
21. Arquilla, John. 2010. "The New Rules of War." *Foreign Policy* 24. February (178): 60–67. doi:10.2307/20684990.
22. US Department of Defense. 2010. "Irregular Warfare : Countering Irregular Threats Joint Operating Concept."
23. Clough, Bruce T. 2002. "Metrics, Schmetrics! How The Heck Do You Determine A UAV's Autonomy Anyway." *2002 Performance Metrics for Intelligent Systems Workshop*. http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA515926.
24. Dekker, Anthony. 2008. "A Taxonomy of Network Centric Warfare Architectures." *Systems Engineering/Test and Evaluation*, 1–14. http://members.ozemail.com.au/~dekker/SETE.Dekker.final.pdf.
25. Weiskopf, Frank, Tim Gion, Dale Elkiss, Harold Gilreath, Jim Bruzek, Robert Bamberger, Ken Grossman, and Jordan Wilkerson. "Control of Cooperative, Autonomous Unmanned Aerial Vehicles." In *Proceedings of First AIAA Technical Conference and Workshop on UAV, Systems, Technologies, and Operations*. 2002.
26. Parunak, H. Van Dyke. 2003. "Making Swarming Happen." In *Conference on Swarming and C4ISR, Tysons Corner, VA 3JAN2003*.
27. Gerkey, Brian P., Mataric, Maja J. 2004. "A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems." The International Journal of Robotics Research 23 (9): 939–54. doi:10.1177/0278364904045564.

# References contd.


28. Dudek, Gregory, Michael R M Jenkin, Evangelos Milios, and David Wilkes. 1993. “A Taxonomy for Multi-Agent Robotics.” *Autonomous Robots* 29.
29. Brambilla, Manuele, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. 2013. “Swarm Robotics: A Review from the Swarm Engineering Perspective.” *Swarm Intelligence* 7 (1): 1–41. doi:10.1007/s11721-012-0075-2.
30. Brooks, Rodney A. "A robust layered control system for a mobile robot. "*Robotics and Automation, IEEE Journal of* 2, no. 1 (1986): 14-23.
31. Brambilla, Manuele, Carlo Pinciroli, Mauro Birattari, and Marco Dorigo. 2012. “Property-Driven Design for Swarm Robotics.” *AAMAS 2012 International Conference on Autonomous Agents and Multiagent Systems*, no. June: in – press. http://iridia.ulb.ac.be/~mbrambilla/pdf/Bra-aamas2012.pdf.
32. Coppin, Gilles, and Francois Legras. 2012. “Autonomy Spectrum and Performance Perception Issues in Swarm Supervisory Control.” *Proceedings of the IEEE* 100 (3): 590–603. doi:10.1109/JPROC.2011.2174103.
33. Goldman, Robert P, Christopher A Miller, Harry B Funk, and John Meisner. 2005. “Optimizing to Satisfice : Using Optimization to Guide Users 1 Introduction.” *Interfaces*, no. January: 18–20.
34. Simmons, Reid, Apfelbaum, David, Fox, Dieter, Goldman, Robert P., Zita Haigh, Karen, Musliner, David J., Pelican, Michael, Thrun, Sebastian. 2000. “Coordinated Deployment of Multiple, Heterogeneous Robots.” *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 127–40. doi:10.1007/978-3-319-10969-5.
35. Chung, Timothy H. 2015. “Dr. Chung Directed Study Notes1.”
36. Matarić, M J. 1995. “Issues and Approaches in the Design of Collective Autonomous Agents.” *Robotics and Autonomous Systems* 16(2-4) (January 1994): 321–31.
37. Senanayake, Madhubhashi, Ilankaikone Senthooran, Jan Carlo, and Hoam Chung. 2015. “Search and Tracking Algorithms for Swarms of Robots : A Survey.” *Robotics and Autonomous Systems*. Elsevier B.V. doi:10.1016/j.robot.2015.08.010.
38. Dudek, Gregory, and Michael Jenkin. *Computational principles of mobile robotics*. Cambridge university press, 2010.
39. Mitchell, Melanie. 2009. *Complexity: A Guided Tour*. Oxford University Press.
40. Reynolds, Craig W. 1987. "Flocks, herds and schools: A distributed behavioral model." In *ACM SIGGRAPH computer graphics*, vol. 21, no. 4, pp. 25-34.
41. Whitcomb, C., Giammarco, K. and Hunt, S., 2015. *An instructional design reference mission for search and rescue operations*. Monterey, California. Naval Postgraduate School.

# References contd.


42. Tritten, James J. 1994. "Naval Perspectives for Military Doctrine Development." doi:10.1017/CBO9781107415324.004.
43. Chung, Timothy H, Kevin D Jones, Michael A Day, Marianna Jones, and Michael Clement. 2015. "50 vs. 50 By 2015: Swarm vs. Swarm UAV Live - Fly Competition At the Naval Postgraduate School," 1–20.
44. Hughes, Wayne P. 2000. Fleet Tactics and Coastal Combat. Edited by U.S. Naval Institute. 2nd ed. Annapolis: U.S. Naval Institute.
45. Zhan, Zhi-Hui, Jun Zhang, Yun Li, and Henry Shu-Hung Chung. 2009. "Adaptive Particle Swarm Optimization." *IEEE Transactions on Systems, Man, and Cybernetics. Part B, Cybernetics : A Publication of the IEEE Systems, Man, and Cybernetics Society* 39 (6): 1362–81. doi:10.1109/TSMCB.2009.2015956.
46. Zweibelson, Ben. 2015. "Let me tell you about the birds and the bees: swarm theory and military decision making." Canadian Military Journal.
47. Wright, David R. 2005. "Finite State Machines." Carolina State University. http://www4.ncsu.edu/~drwrigh3/docs/courses/csc216/fsm-notes.pdf.
48. Rodano, Matthew, and Kristin Giammarco. "A formal method for evaluation of a modeled system architecture." *Procedia Computer Science* 20 (2013): 210-215.
49. DARPA TTO. 2017. "Broad Agency Announcement Offensive Swarm Enabled Tactics (OFFSET) Tactical Technology Office (TTO)." Arlington, VA: DARPA TTO.
50. Sheard, Sarah, Stephen Cook, Eric Honour, Duane Hybertson, Joseph Krupa, Jimmie Mcever, Dorothy Mckinney, et al. 2015. "A Complexity Primer for Systems Engineers." *INCOSE Complex Systems Working Group*, no. July: 1–17.
51. Cross, Nigel, and Robin Roy. 1989. *Engineering Design Methods*. 4th Ed. New York: Wiley.
52. Clements, Paul C, Rick Kazman, and Mark Klein. 2001. "Evaluating Software Architectures: Methods and Case Studies." In *Evaluating Software Architectures: Methods and Case Studies*, 19–42. London: Addison-Wesley Professional.
53. Auguston, Mikhail. 2017. "System and Software Architecture and Workflow Modeling Language Manual." Monterey, CA: Naval Postgraduate School.

# BACK UP

# Background- Enabling Technology

- Swarm technology – inspired by biology:
  - —Swarm systems are robust, flexible, scalable
  - —Emergent behavior arises from interactions between agents

- Enabling technologies for UAV swarms:
  - —Improved communication networks including meshed ad-hoc networks
  - —Cost-effective miniaturized electronics: GPS, video cameras, radio receivers, autopilot processors
  - —Automation - must shift from operators to monitors and supervisors

www.wired.com

www.wikipedia.org

Reference: Sahin 2005

# Background- Swarm C2 Architectures

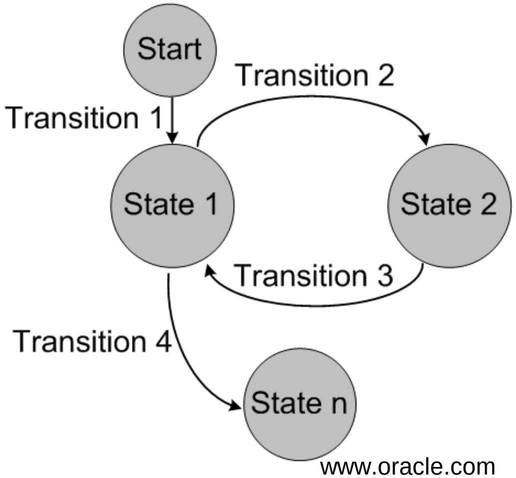


- *Orchestrated*- one agent selected as temporary leader based on specified factors (e.g., location, state, mission scenario)
  - Architecture is somewhat robust, but not scalable to large or geographically dispersed swarms, and places significant processing burden on one agent
- *Hierarchical*– resembles traditional military command and control (C2)
  - Simplifies data flow, but not robust and inflexible when dealing with dynamic situations that require rapid reactions from agents
- *Distributed* - characterized by absence of leader; swarm decisions made via collective consensus among agents
  - Robust and scalable, but requires communication network that will support potentially increased data traffic, such as wireless, mesh communication networks
- *Emergent swarming*- describes relationships which occur in ant, termite, and bee colonies in which there is no management
  - Agents have no leader, have low situational awareness, and follow simple rules based on local information (i.e. sharing pheremone signals)
  - Have potential to become more relevant as genetic algorithms are further developed

References: Dekker 2008, Chung et al. 2013

# Background- Architecting a Swarm

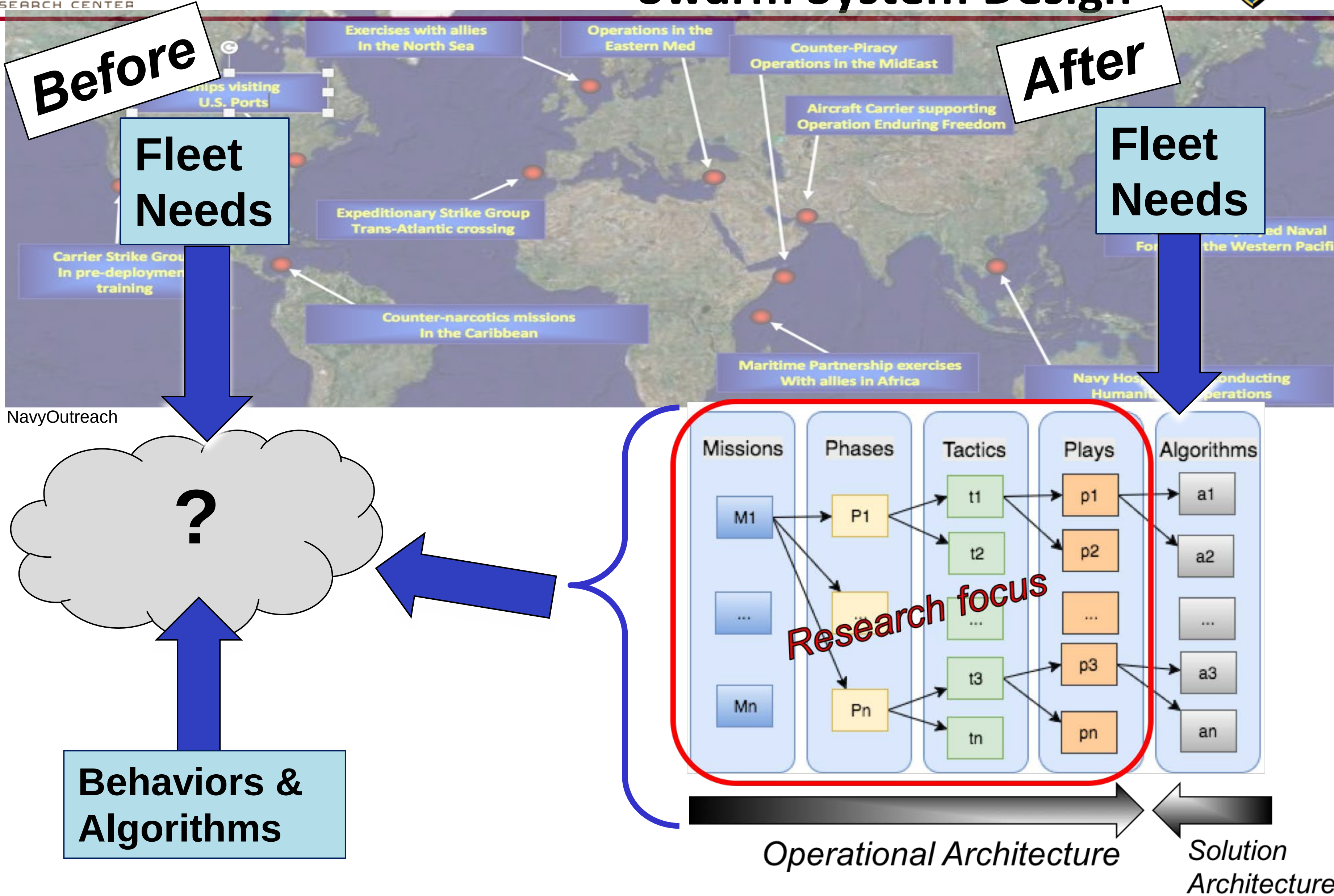
- Hybrid C2 architectures can be used to maximize strengths of each:
  - US Navy's Cooperative Engagement Capability (CEC) anti-air warfare system utilizes a distributed architecture for situational awareness data and an orchestrated architecture for target selection
- Finite State Machines (FSM):
  - Used in modeling multi-vehicle autonomous, unmanned system architectures
  - Applicable to military swarm systems performing high risk missions
  - Probabilistic FSMs can be used to allow for bounded behavior variability
- Petri Nets:
  - Effective in visualizing and analyzing systems in which there are multiple, independent activities occurring at same time



www.oracle.com

References: Dekker 2008, Weiskopf et al., 2002, Zhu et al. 2009

# Problem Space Examination-
# Swarm System Design






Reference: DARPA OFFSET BAA, 2017

# Background- Doctrine

- Military doctrine
  - “Fundamental principles by which the military forces or elements thereof guide their action in support of national objectives” *(JP 1-02)*
  - Influenced by technology, the enemy’s capabilities, organizational structure, and geography
  - Applies at every level of warfare (strategic, operational, tactical)

- Military tactics
  - Handling of forces in battle
  - “The sum of the art and science of the actual application of combat power” *(Arthur Cebrowski, VADM USN, ret)*
  - “…the choice of tactics will also be governed by scouting effectiveness and weapons range” *(Hughes)*
  - Tactical doctrine organizes the playbook

*“Fire effectively first!” –Wayne Hughes, CAPT USN (ret)*


References: Arquilla 2000, Hughes 2000, Tritten 1994

# Background- Swarming Doctrine

- Swarming origins:
  - —British vs. Spanish Armada in 1588
  - —British vs. swarming German U-boat wolf packs in N. Atlantic Japanese kamikaze attacks against US Navy
  - —Al Qaeda's strikes on multiple US targets on 11 Sept. 2001
  - —Typical NGO operations

- What will modern swarming doctrine look like?
  - —Transition from “few and large” forces to “many and small” units
  - —Centralized strategy
  - —Widely distributed, smaller units executing pulse-like tactics
  - —Distributed Lethality?

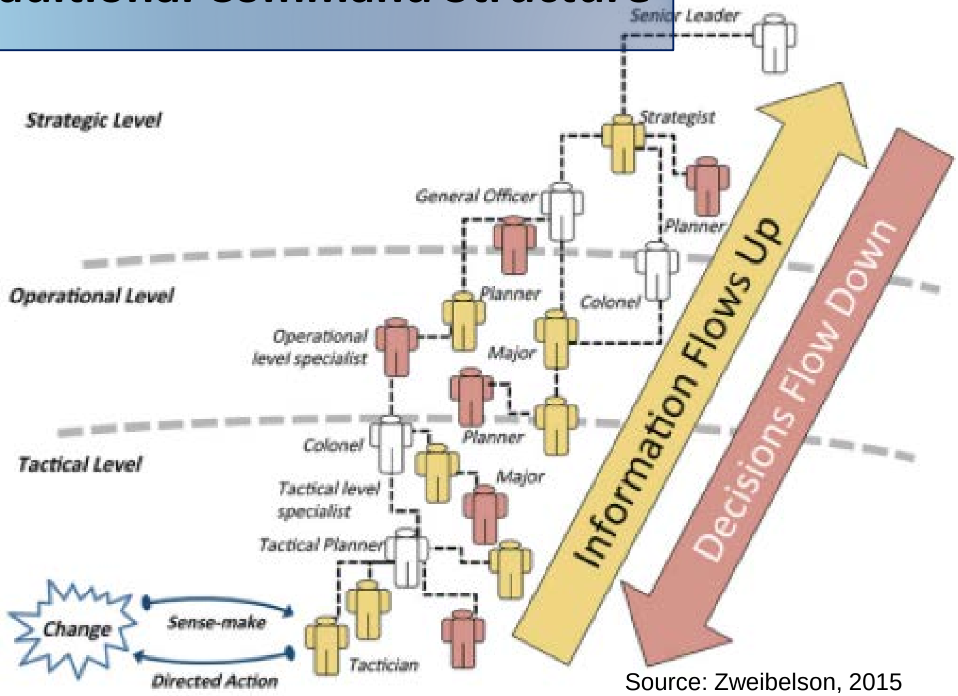


References: Arquilla 1997, 2000

# Background- Current Doctrine vs. Swarming Doctrine



## Traditional Command Structure

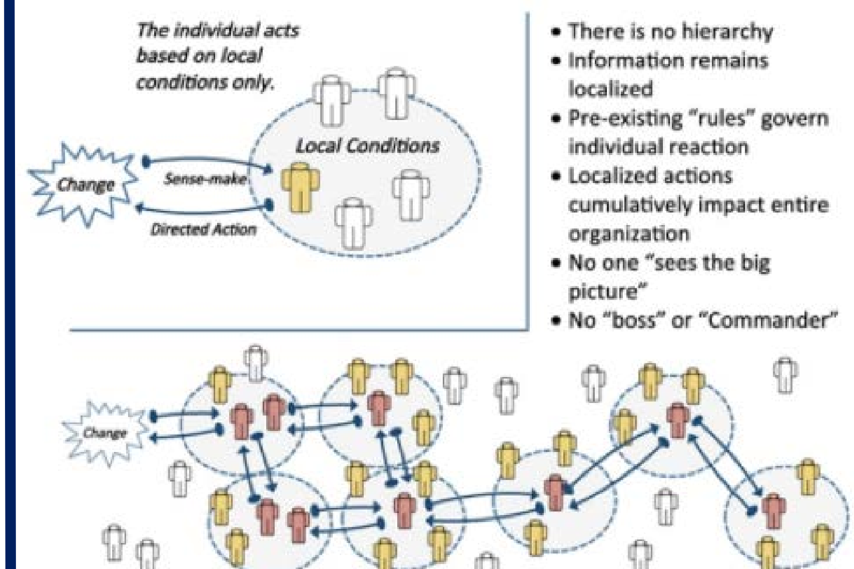



Source: Zweibelson, 2015

- Hierarchical
- Carrier strike group, amphibious strike group

## Swarming Command Structure?

- Widely distributed, small units, multi-axis, convergent attacks
- Disperse and amass
- Historical: German U-boats, Japanese kamikaze, Al Qaeda



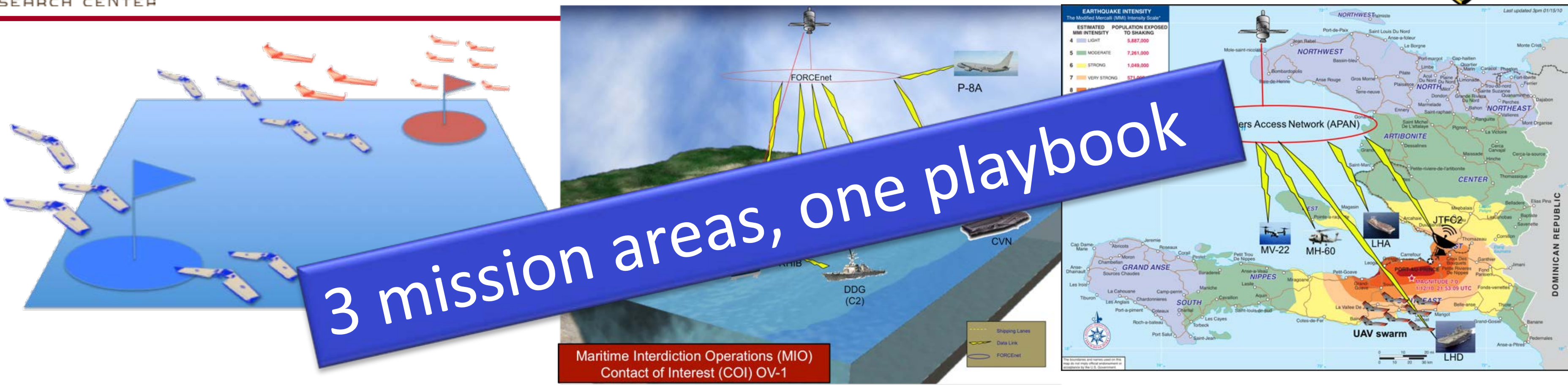

Source: Zweibelson, 2015

***Transition from "few and large" forces to "many and small"*** **-Dr. Arquilla**

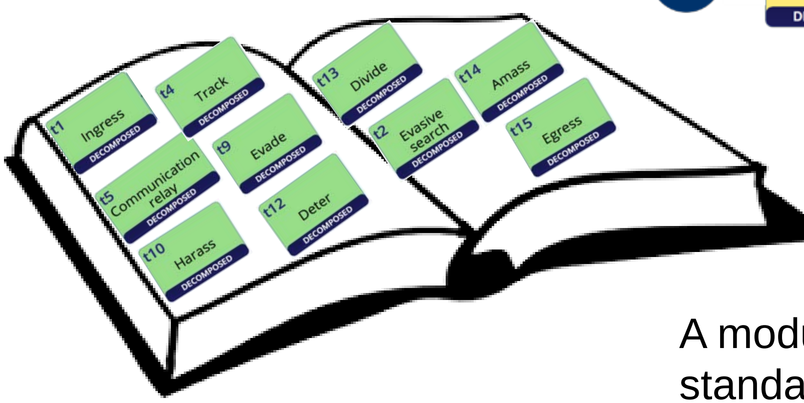


References: Edwards 2010, Arquilla 2000

# Research Objectives




Common conceptual architecture-level patterns for mission-suitable swarm systems across a range of missions


A modular, mission-oriented playbook from which standard swarm tactics and missions can be formulated

# Research Approach

## Problem Space Examination

- Review swarm system literature
- Study current UAV swarm operations
- Identify research opportunities

## Solution Generation

Develop:

- Swarm architecture
- Mission scenarios
- Operational activity models
- Logic checking models

## Solution Evaluation

- Logic checks
- Evaluate system behavior
- Elicit feedback from HSR
- Evaluate for intuitiveness, modularity, and integration of mission doctrine

## Implication of Solution

- Document architecture and methodology
- Demonstrate reusability of tactics and plays across missions
- Demonstrate composability at each level

Reference: Cross 1989

# Related Work



- Dudek's swarm robotics taxonomy (Dudek et al., 1993)
- Bottom-up, behavior-based design
  - Agent-based modeling (Bonabeau 2002, Munoz 2011)
  - Finite state machine (automata) (Weiskopf 2002, Soyal & Sahin 2005)
  - Brooks Subsumption architecture (Brooks 1985)
  - Petri Nets (Levis & Wagenhals 2000, Palamara 2008)
- Top-down design methods
  - DeLoach et al.'s Multi-agent Systems Engineering methodology (DeLoach et al. 2001)
  - Brambilla's property-driven, four phase method (Brambilla et al., 2012)
- Playbooks
  - RoboFlag multi-vehicle simulation environment (Parasuraman 2003, Squire et al. 2004)
  - RoboCup soccer (Browning et al. 2004)
  - McLurkin's library of behaviors for swarm robots (McLurkin 2004)
  - Smart Information Flow Technology (SIFT) Playbook-enhanced Variable Autonomy Control System (PVACS) (Goldman 2005)
  - DARPA OFFSET program - human-swarm teaming and swarm autonomy within an urban gaming environment (DARPA TTO 2017)

# Solution Generation-Heuristics for Model Building

- Applied as guidelines to Innoslate models and simulation:
  - Every activity not designated a context activity should have at least one parent

    $$(\forall a_1 \in A)[\neg context(a_1) \rightarrow (\exists a_2 \in A) decomposes(a_1, a_2)]$$

  - No activity shall have exactly one child

    $$(\forall a_1 \in A)\ (\forall a_2 \in A)\ [decomposed\ by(a_1, a_2) \rightarrow (\exists a_3 \in A) decomposedby(a_1, a_3) \wedge (a_2 \neq a_3)]$$

  - No activity shall be decomposed by itself

    $$(\forall a \in A)[\neg decomposedby(a, a)]$$

  - Every activity shall have at least one input or trigger

    $$(\forall a \in A)\ (\exists r \in R)\ [input(r, a) \vee trigger(r, a)]$$

  - No performer shall have more than seven children

    $$(\forall p_1 \in P)\ [(\forall p_2 \in P)|decomposedby(p_1, p_2)| \leq 7]$$

Source: Rodano & Giammarco 2013

# Solution Generation & Evaluation – Mission as FSM

- Finite state machines are concise way to depict swarm behavior
  - Specify each tactic as a state
  - Sub-swarms operate in one state at a time
- A finite state machine (or automaton) *M*, can be defined by a 5-tuple[1]:

$$(\mathcal{E}, \mathcal{S}, s_o, \mathrm{F}, \delta)$$

wherein:

- $\mathcal{E}$ is the set of inputs to *M*
- $\mathcal{S}$ is the set of states, including tactics, of *M*
- $s_o \in \mathcal{S}$ is the initial state of *M* (preflight completed and flight ready)
- $\mathrm{F} \subseteq \mathcal{S}$ is the final state of *M* (all UAVs recovered)
- $\delta: \mathcal{S} \times \mathcal{E} \Rightarrow \mathcal{S}$ is the transition function (mappings of inputs to original states which result in state change)

- FSM has modelling implications in Innoslate and Monterey Phoenix
  - Innoslate FSM do not interface with simulation
  - MP does not permit implicit or explicit recursion in grammar rules[2]


Reference: [1]Wright 2005, [2]Auguston 2017

# Solution Evaluation- Modularity of Plays Across Missions

| Play | SvS | MIO | MIO HSR | HADR |
|---|---|---|---|---|
| Launch | B | B | 8 | B |
| Transit to WP | B | B | 8 | B |
| Orbit | | B | 7 | B |
| Racetrack | | B | 4 | B |
| Split (logic based) | | B | 7 | B |
| Join | B | B | 8 | B |
| Disperse | B | B | 8 | |
| Sensors ON | B | B | 8 | B |
| Sensors OFF | B | B | 7 | B |
| Sensors EMCON | | B | 2 | |
| Transmit video | | B | 8 | B |
| Terminal approach | B | B | 7 | B |
| Landing | B | B | 8 | B |
| Ladder pattern | | B | 2 | B |
| Expanding square pattern | B | B | 2 | B |
| Constricting square pattern | | B | 2 | B |
| Grid pattern | | B | 2 | B |
| Random pattern | B | | 3 | |
| Weapon armed | B | | | |
| Weapon fire | B | | 1 | |
| Follow target | | B | 5 | B |
| Forward communication | | B | 4 | B |
| Jam | | | 1 | |
| Smart greedy shooter | B | | 1 | |
| Patrol box shooter | B | | | |
| Wingman shooter | B | | | |
| Tail following | B | | | |
| Option | | | 5 | |

B = selected for baseline mission case study (Innoslate model)

# = number of HSR participants who selected play

# Solution Generation & Evaluation – MIO Mission Composition

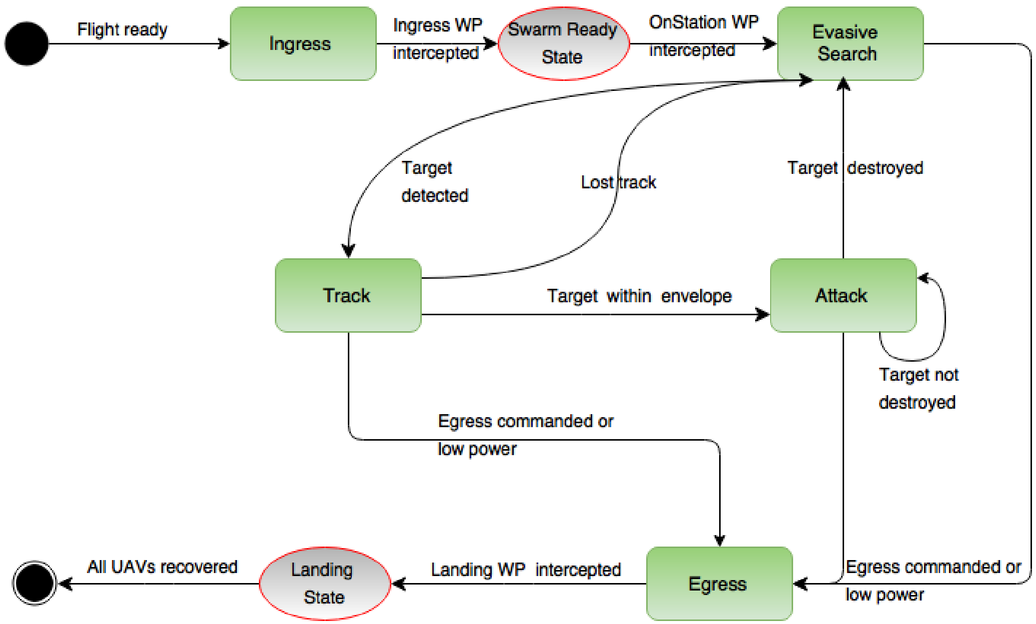
| Tactic ID | Tactic Name | Play ID | Play Name | Algorithm ID | Algorithm Name |
|---|---|---|---|---|---|
| t1 | Ingress | p8.2 | Sensors OFF | a2 | Scheduling |
| | | p1 | Launch | a3 | Sorting |
| | | p8.1 | Sensors ON | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | p2 | Transit to waypoint | a1 | Flocking |
| | | | | a1.1 | Boid's |
| | | | | a1.2 | Particle Swarm Optimization (PSO) |
| t13 | Divide | p5 | Split (logic-based) | a3 | Sorting |
| | | p8.1 | Sensors ON | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | p3 | Orbit | a1 | Flocking |
| | | | | a1.1 | Boid's |
| | | | | a1.2 | PSO |
| t3 | Efficient search | p8.1 | Sensors ON | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | p10 | Ladder pattern | a1 | Flocking |
| | | | | a1.1 | Boid's |
| | | | | a1.2 | PSO |
| t8 | Monitor | p8.1 | Sensors ON | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | p4 | Racetrack | a1 | Flocking |
| | | | | a1.1 | Boid's |
| | | | | a1.2 | PSO |
| t5 | Communication relay | p8.1 | Sensors ON | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | p8.4 | Transmit video | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | p13 | Grid pattern | a1.3 | Levy flight |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | | | a8.7 | Brownian motion |
| | | p18 | Forward communication | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| t4 | Track | p8.1 | Sensors ON | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | p17 | Follow target | a6 | Firefly algorithm |
| | | | | a8.3 | Ant colony optimization |
| | | | | a5 | Artificial potential fields |
| t9 | Evade | p8.3 | EMCON | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | p6 | Join | a3 | Sorting |
| | | | | a5 | Artificial potential fields |
| | | p7 | Disperse | a5 | Artificial potential fields |
| t14 | Amass | p6 | Join | a4 | Collective consensus |
| | | | | a5 | Artificial potential fields |
| | | p8.1 | Sensors ON | a2 | Scheduling |
| | | | | a3 | Sorting |
| | | | | a4 | Collective consensus |
| | | p3 | Orbit | a1 | Flocking |
| | | | | a1.1 | Boid's |
| | | | | a1.2 | PSO |
| t15 | Egress | p2 | Transit to waypoint | a1 | Flocking |
| | | | | a1.1 | Boid's |
| | | | | a1.2 | PSO |
| | | p9.1 | Terminal approach | a3 | Sorting |
| | | p9.2 | Landing | a3 | Sorting |
| | | p8.2 | Sensors OFF | a2 | Scheduling |

tactics

plays

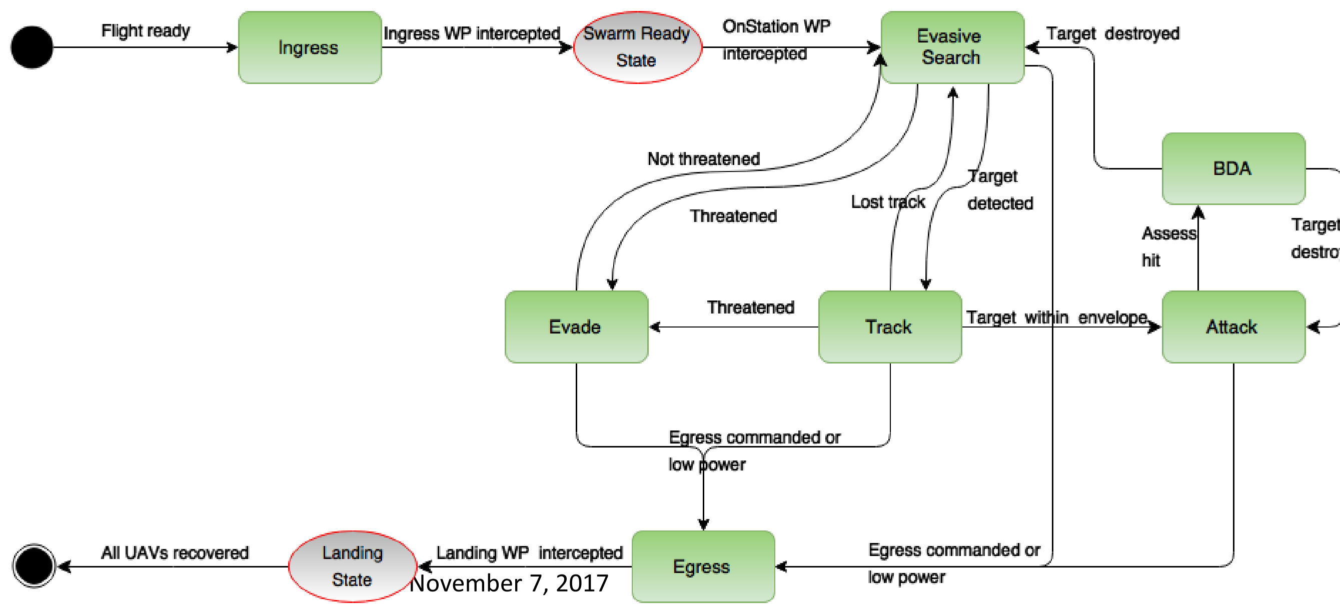
algorithms

# Solution Generation & Evaluation – SvS Mission as FSM

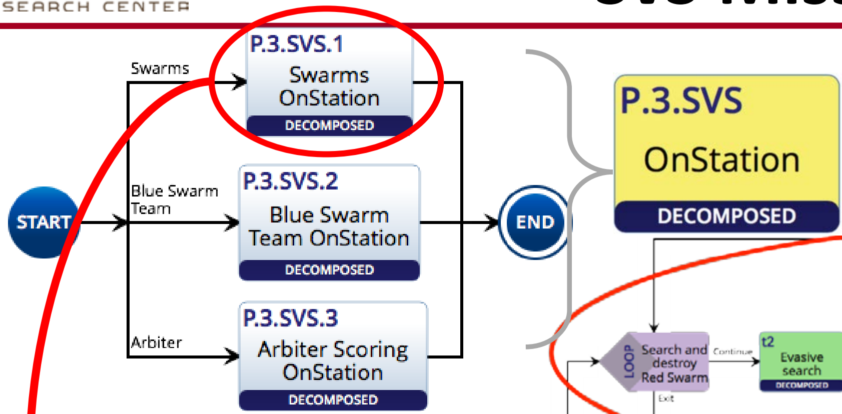



FSM applied to field experimentation scenario

Modified with additional MASC swarm tactics



# Solution Evaluation – SvS Mission Innoslate Simulation




SvS Mission at Tactics Level using MASC Framework

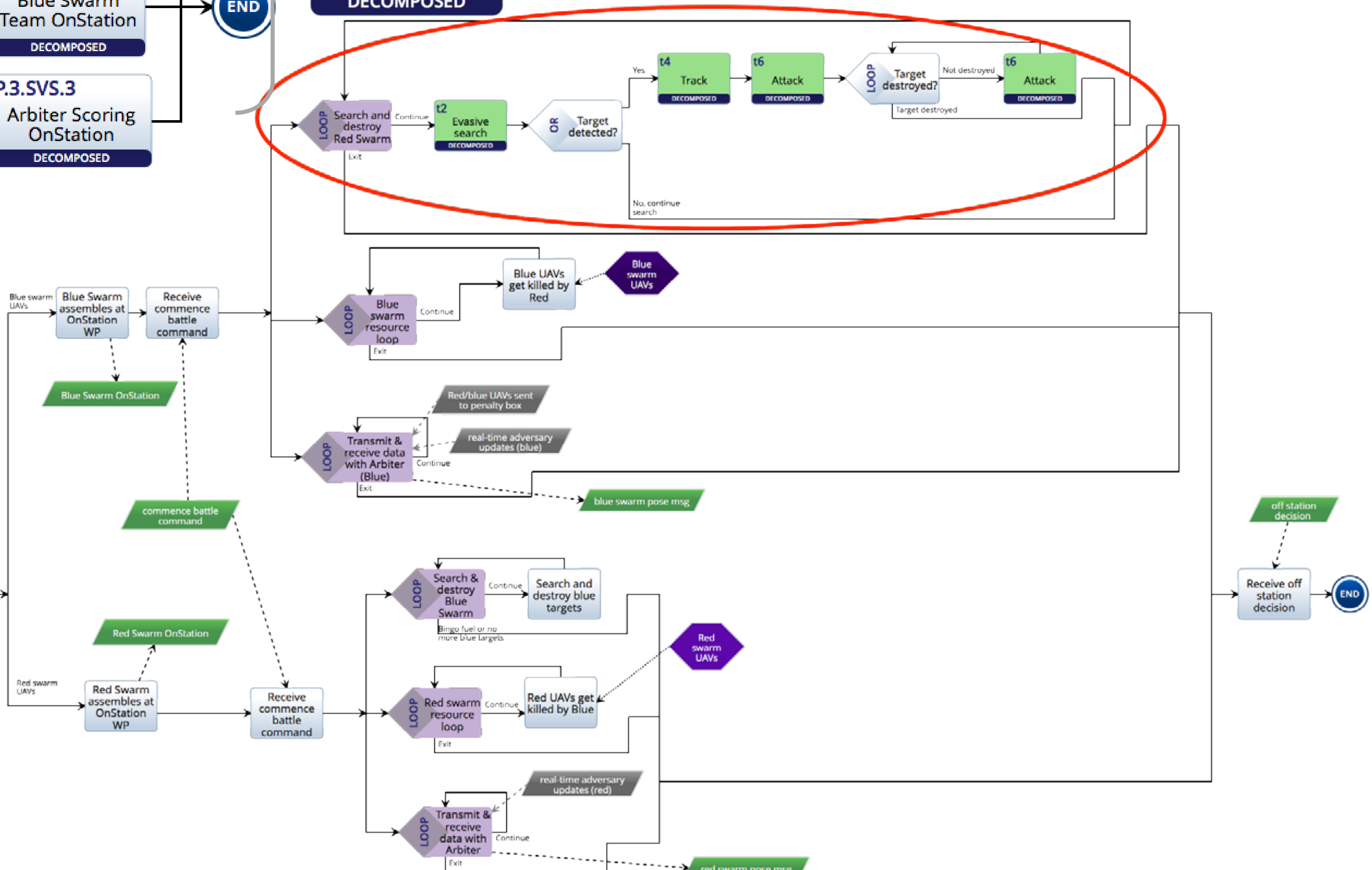
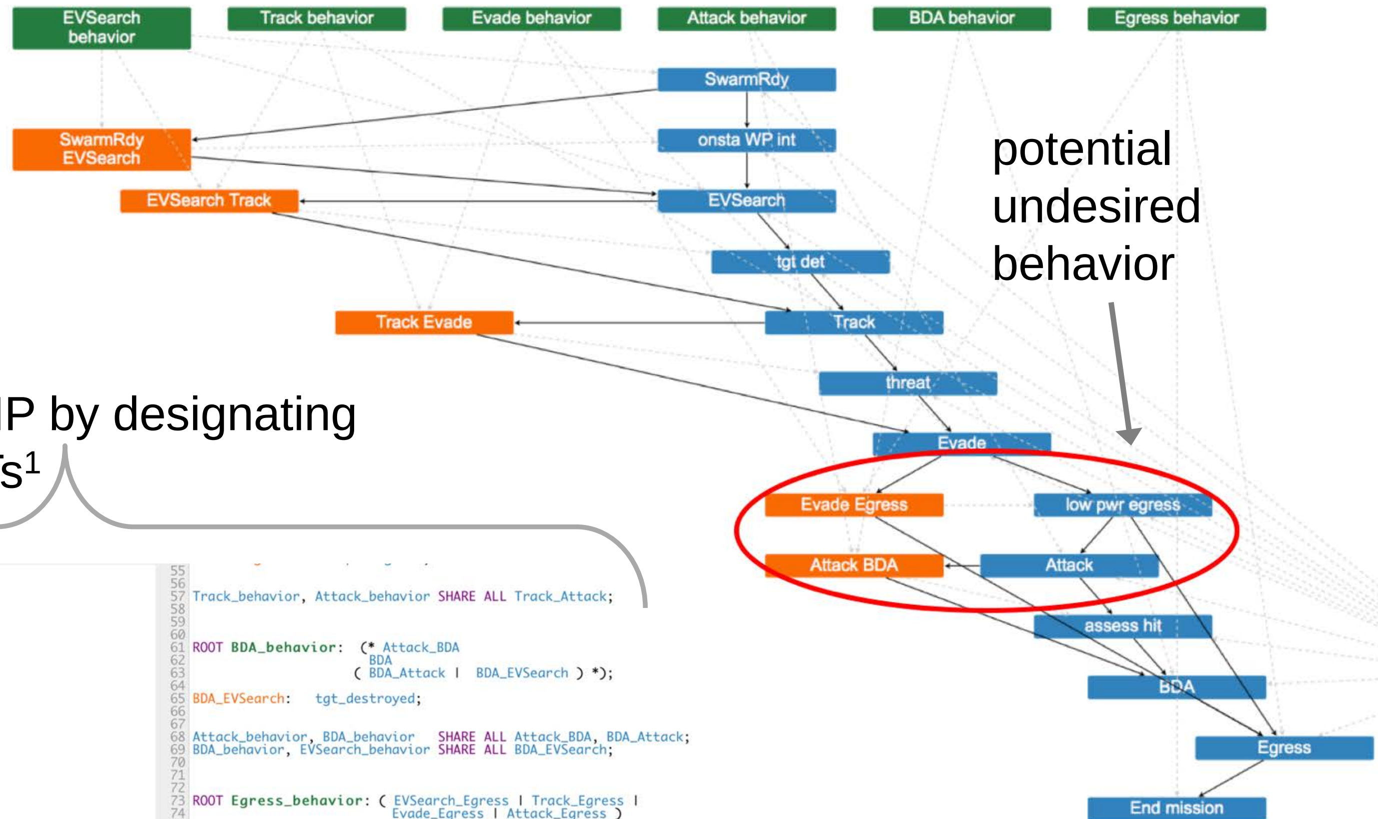



Diagram of part of simulation developed in *Innoslate*™

# Solution Evaluation – SvS Mission as FSM in Monterey Phoenix




FSM converted to MP by designating the tactics as ROOTs[1]

```
/* K. Giles 09AUG2017
swarm vs. swarm FSM model to MP conversion with names,
simplified  to SVS onstation tactics as states only
Based on Dr.A's microwave example, dfa_2.
-----------------------------------------*/
SCHEMA SVS_FSM

ROOT EVSearch_behavior:   SwarmRdy
                          SwarmRdy_EVSearch
                          EVSearch
                   (* (   EVSearch_Track  |
                          EVSearch_Evade  |
                          EVSearch_Egress  )*);

SwarmRdy_EVSearch:  onsta_WP_int;
EVSearch_Track:     tgt_det;
EVSearch_Evade:     threat;
EVSearch_Egress:    low_pwr_egress;

ROOT Track_behavior:      EVSearch_Track
                          Track
                   (* (   Track_EVSearch |
                          Track_Evade    |
                          Track_Attack   |
                          Track_Egress  )*);

Track_EVSearch: lost_track;
Track_Evade:    threat;
Track_Attack:   tgt_in_range;
Track_Egress:   low_pwr_egress;

EVSearch_behavior, Track_behavior SHARE ALL EVSearch_Track, Track_EVSearch;


ROOT Evade_behavior:  (* ( Track_Evade | EVSearch_Evade )
                             Evade
                      (  Evade_EVSearch  | Evade_Egress ) *);

Evade_EVSearch:     no_threat;
Evade_Egress:       low_pwr_egress;

Evade_behavior, Track_behavior      SHARE ALL Track_Evade;
EVSearch_behavior, Evade_behavior   SHARE ALL EVSearch_Evade, Evade_EVSearch;


ROOT Attack_behavior: (*( Track_Attack  |
                          BDA_Attack )
                          Attack
                       (  Attack_BDA | Attack_Egress )*);

BDA_Attack:     tgt_not_destroyed;
Attack_BDA:     assess_hit;
Attack_Egress:  low_pwr_egress;


Track_behavior, Attack_behavior SHARE ALL Track_Attack;



ROOT BDA_behavior:  (* Attack_BDA
                       BDA
                     ( BDA_Attack |  BDA_EVSearch ) *);

BDA_EVSearch:   tgt_destroyed;


Attack_behavior, BDA_behavior   SHARE ALL Attack_BDA, BDA_Attack;
BDA_behavior, EVSearch_behavior SHARE ALL BDA_EVSearch;



ROOT Egress_behavior: ( EVSearch_Egress | Track_Egress |
                        Evade_Egress | Attack_Egress )
                        Egress;


EVSearch_behavior, Egress_behavior  SHARE ALL EVSearch_Egress;
Track_behavior, Egress_behavior     SHARE ALL Track_Egress;
Evade_behavior, Egress_behavior     SHARE ALL Evade_Egress;
Attack_behavior, Egress_behavior    SHARE ALL Attack_Egress;


ROOT Path:  ;       /* Path is container for path sequence*/

COORDINATE  $a:     ( SwarmRdy | EVSearch | Track | Evade |
                      Attack | BDA | Egress |
                      onsta_WP_int | tgt_det | lost_track |
                      low_pwr_egress | threat | no_threat |
                      assess_hit | tgt_destroyed |
                      tgt_not_destroyed | tgt_in_range )
    DO
        ADD $a IN Path;
    OD;

COORDINATE  <SORT CUT_END>      $a: $$EVENT SUCH THAT $a IN Path,
            <SORT CUT_FRONT>    $b: $$EVENT SUCH THAT $a IN Path
    DO
        ADD $a PRECEDES $b;
    OD;

/*---------------------------
    ASSERTION CHECKING
---------------------------*/
ENSURE FOREACH $x:low_pwr_egress
    #Track AFTER $x ==0;
```

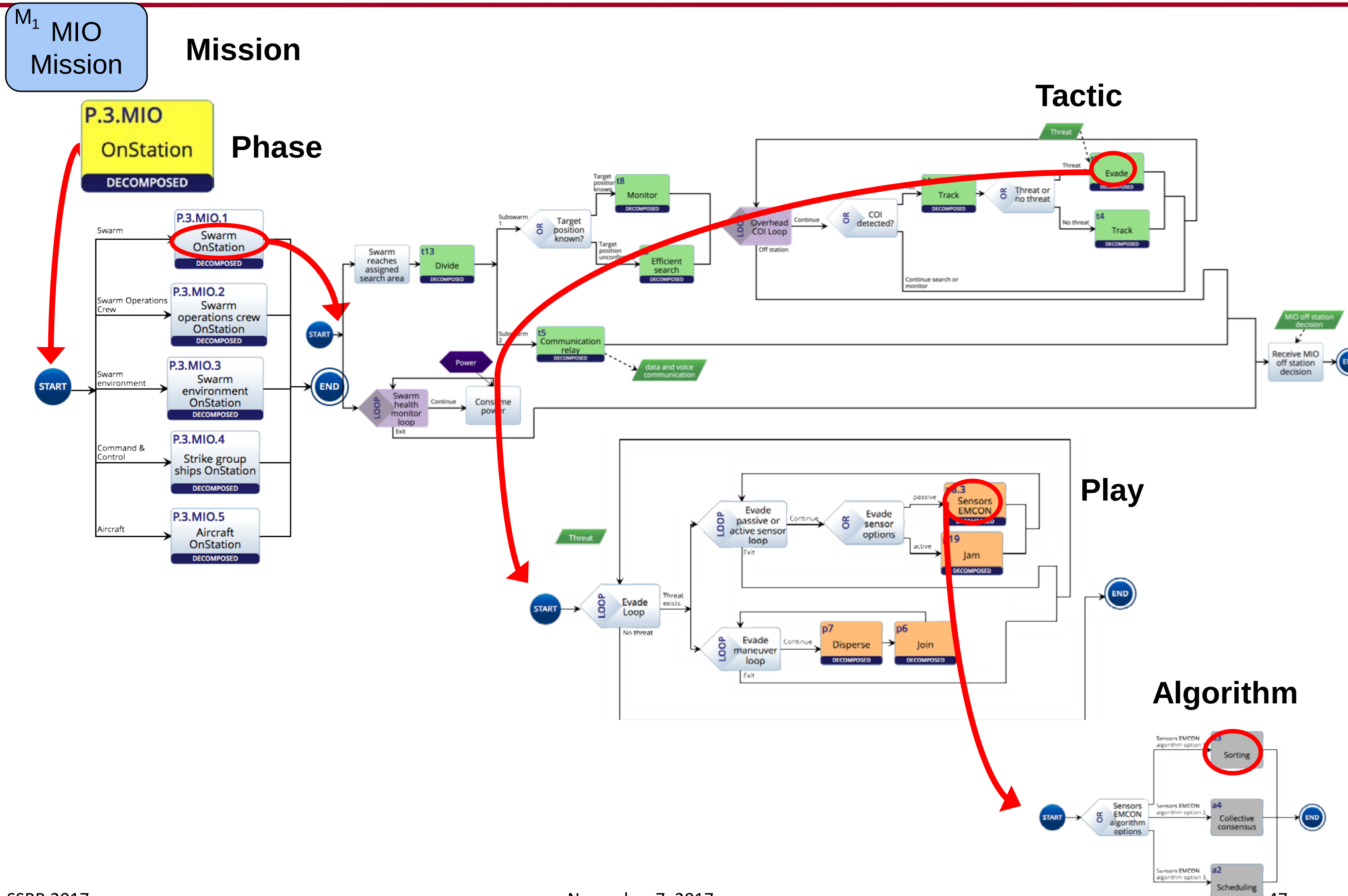
Reference: [1] Auguston 2017

# Solution Evaluation- Modularity of Tactics Across Mission

| Tactic | SvS | MIO | MIO HSR | HADR |
|---|---|---|---|---|
| Ingress | B | B | 7 | B |
| Evasive search | B | | 4 | |
| Efficient search | | B | 5 | B |
| Track | B | B | 7 | B |
| Communication relay | | B | 7 | B |
| Attack | B | | | |
| BDA | B | | 3 | |
| Monitor | | B | 7 | B |
| Evade | B | B | 7 | |
| Harass | | | 3 | |
| Defend | | | 1 | |
| Deter | | | 2 | |
| Divide | | B | 7 | B |
| Amass | | B | 6 | B |
| Egress | B | B | 6 | B |
| ACM | | | | |
| Option | | | 1 | |

B = selected for baseline mission case study (Innoslate model)
# = number of HSR participants who selected tactic

# Solution Evaluation- Composability



SYSTEMS ENGINEERING RESEARCH CENTER

# Research Scope & Assumptions

NPS PRAESTANTIA PER SCIENTIAM 1909

Not developing new swarm behavior algorithms

Not including system failures or failsafe modes in models

Subset of US Navy Missions

Tactics are focused on inflight mission phases

DoD category 1-2 fixed-wing, homogeneous UAVs

System operates under distributed control using mesh ad-hoc network

Not developing collision avoidance algorithms

Not addressing operational failures in models

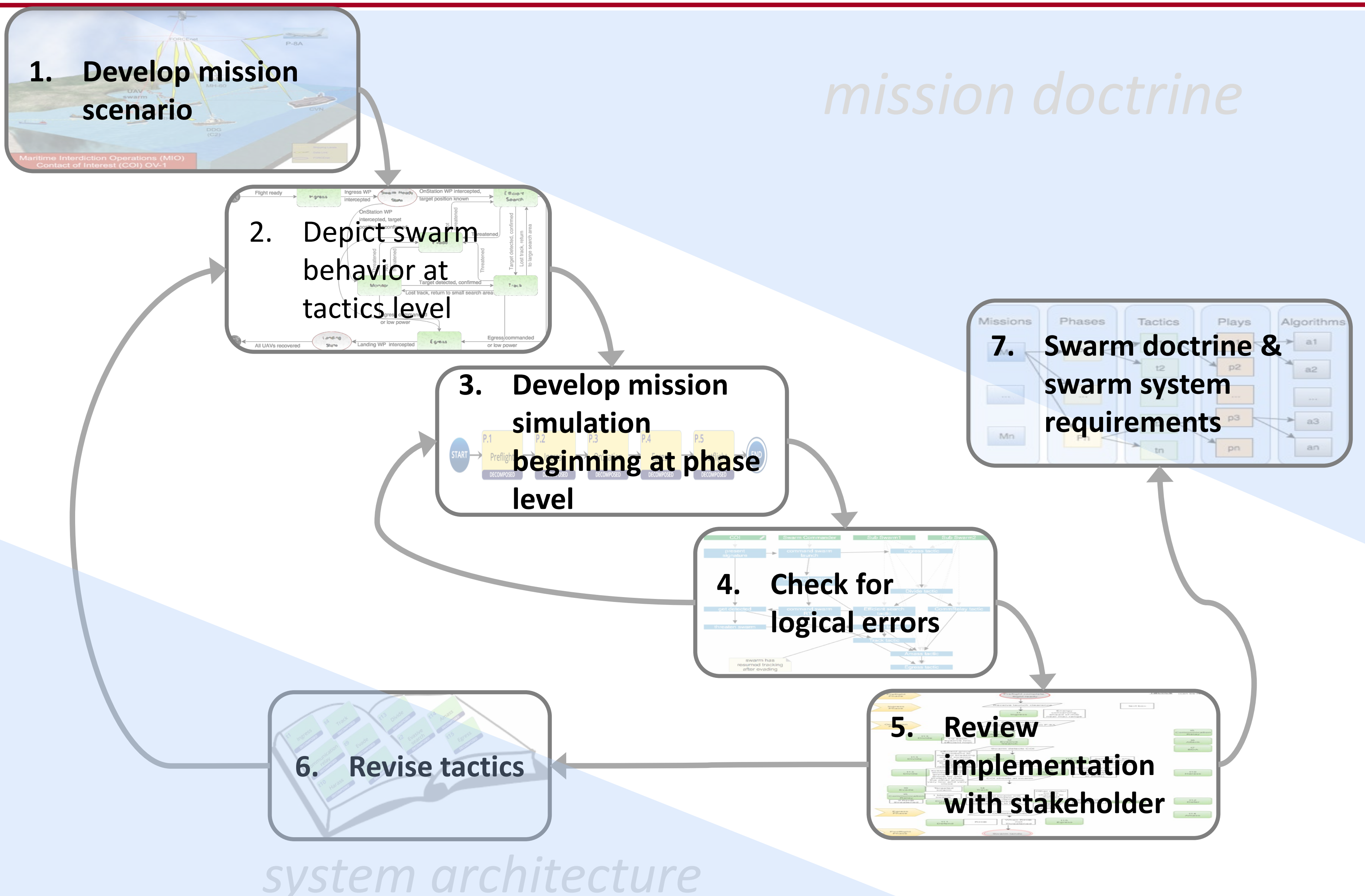
Swarm Operations Team: Swarm Commander, Swarm Health Monitor, Ground Crew

Sub-swarm is lowest level unit for tactics employment

Each sub-swarm executes 1 tactic at a given time

Not focusing on logistical activities

Not developing game-like prototype (yet)



mission doctrine
1. Develop mission scenario
2. Depict swarm behavior at tactics level
3. Develop mission simulation beginning at phase level
4. Check for logical errors
5. Review implementation with stakeholder
6. Revise tactics
7. Swarm doctrine & swarm system requirements
Missions
Phases
Tactics
Plays
Algorithms
system architecture